



TUGAS AKHIR – TF 141581

**PREDIKSI CUACA UNTUK PENINGKATAN
KESELAMATAN PENERBANGAN DENGAN
MENGUNAKAN JARINGAN SARAF
TIRUAN DAN *PARTICLE SWARM*
*OPTIMIZATION***

Evandino Febrianza Hartantyo
NRP 2413 100 146

Dosen Pembimbing
Dr.Ir. Syamsul Arifin, MT

DEPARTEMEN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT – TF 141581

***WEATHER PREDICTION TO INCREASE THE
FLIGHT SAFETY BY USING ARTIFICIAL
NEURAL NETWORK AND PARTICLE SWARM
OPTIMIZATION***

Evandino Febrianza Hartantyo
NRP 2413 100 146

Supervisor
Dr.Ir. Syamsul Arifin, MT

DEPARTMENT OF ENGINEERING PHYSICS
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2017

PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini:

Nama : Evandino Febrianza Hartantyo
NRP : 2413100146
Departemen / Prodi : Teknik Fisika / S1 Teknik Fisika
Fakultas : Fakultas Teknologi Industri
Perguruan Tinggi : Institut Teknologi Sepuluh Nopember

Dengan ini menyatakan bahwa Tugas Akhir dengan judul “Prediksi Cuaca Untuk Peningkatan Keselamatan Penerbangan dengan Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*” adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain. Apabila di kemudian hari terbukti terdapat plagiat pada Tugas Akhir ini, maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.

Surabaya, 21 Juni 2017
Yang membuat pernyataan,



Evandino Febrianza Hartantyo
NRP. 2413100146

LEMBAR PENGESAHAN

PREDIKSI CUACA UNTUK PENINGKATAN KESELAMATAN PENERBANGAN DENGAN MENGUNAKAN JARINGAN SARAF TIRUAN DAN *PARTICLE SWARM OPTIMIZATION*

TUGAS AKHIR

Oleh :

Evandino Febrianza Hartantyo

NRP : 2413100146

Surabaya, Juli 2017
Mengetahui/Menyetujui

Pembimbing

Ketua Jurusan

Fisika FTI-ITS

Dr.Ir.Syamsul Arifin, MT

NIPN. 19630907 198903 1 004

Agus Muhammad Hatta, S.T, M.Si, Ph.D

NIPN. 197809022003121002

LEMBAR PENGESAHAN

PREDIKSI CUACA UNTUK PENINGKATAN KESELAMATAN PENERBANGAN DENGAN MENGUNAKAN JARINGAN SARAF TIRUAN DAN *PARTICLE SWARM OPTIMIZATION*

TUGAS AKHIR


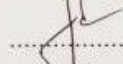
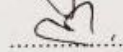
Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
pada
Bidang Studi Instrumentasi
Program Studi S-1 Departemen Teknik Fisika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Oleh :

EVANDINO FEBRIANZA HARTANTYO
NRP. 2413100146

Disetujui oleh Tim Penguji Tugas Akhir :

1. Dr. Ir. Syamsul Arifin, MT
2. Ir. Ya'umar, MT
3. Andi Rahmadiansah, ST, MT

 (Pembimbing)
 (Penguji I)
 (Penguji II)

SURABAYA
JULI, 2017

PREDIKSI CUACA UNTUK PENINGKATAN KESELAMATAN PENERBANGAN DENGAN MENGUNAKAN JARINGAN SARAF TIRUAN DAN *PARTICLE SWARM OPTIMIZATION*

Nama Mahasiswa : Evandino Febrianza Hartantyo
NRP : 2413 100 146
Jurusan : Teknik Fisika FTI-ITS
Dosen Pembimbing : Dr. Ir. Syamsul Arifin, MT

Abstrak

Cuaca memiliki peran yang signifikan pada kecelakaan penerbangan yaitu sebesar 23% dari seluruh penerbangan yang ada. Dalam upaya mengurangi kecelakaan dibutuhkan sebuah metode yang tepat dan akurat dalam melakukan prediksi cuaca, sehingga dapat membantu dalam membuat suatu keputusan pada waktu sebelum penerbangan ataupun saat penerbangan sedang berlangsung. Untuk mengatasi masalah ini dilakukan penelitian mengenai prediksi cuaca dengan menggunakan metode gabungan jaringan saraf tiruan dan *particle swarm optimization* (JST-PSO). Nilai RMSE terbaik menggunakan metode gabungan JST-PSO terdapat pada *hidden node* berjumlah 6 *node* pada prediksi kecepatan angin yaitu sebesar 0,0721 dan 9 *node* pada prediksi visibilitas yaitu sebesar 0,0837. Dapat disimpulkan dari hasil tersebut bahwa metode *Particle Swarm Optimization* dapat menghasilkan bobot yang optimum dan dapat melakukan prediksi cuaca untuk meningkatkan keselamatan penerbangan, dilihat dari nilai RMSE yang dihasilkan metode gabungan hampir sama dengan tanpa menggunakan metode *Particle Swarm Optimization*.

Kata Kunci : Prediksi Cuaca, Jaringan Saraf Tiruan, *Particle Swarm Optimization*

WEATHER PREDICTION TO INCREASE THE FLIGHT SAFETY BY USING ARTIFICIAL NEURAL NETWORK AND PARTICLE SWARM OPTIMIZATION

Name : ***Evandino Febrianza Hartantyo***
NRP : ***2413 100 146***
Department : ***Engineering Physics FTI-ITS***
Supervisor : ***Dr. Ir. Syamsul Arifin, MT***

Abstract

Weather has a significant role in aviation accidents that amounted to 23% of all flights. In an effort to reduce accidents requires a precise and accurate method of weather prediction, so that it can assist in making a decision before the flight or flight time is in progress. To solve this problem, research on weather prediction using a combined method of neural network and particle swarm optimization (ANN-PSO) was conducted. The best RMSE value using ANN-PSO combined method is found in the hidden node of 6 nodes in the predicted wind speed of 0.0721 and 9 nodes in the visibility prediction of 0.0837. It can be concluded from these results that Particle Swarm Optimization method can produce the optimum weight and can do weather prediction to improve flight safety, seen from root mean square error (RMSE) value produced by the combined method is similar to without using Particle Swarm Optimization method.

Keyword : ***Weather Prediction, Artificial Neural Network, Particle Swarm Optimization***

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT, karena atas karunia, hikmat, dan rahmat-Nya sehingga penulis diberikan kesehatan, kemudahan, dan kelancaran dalam menyelesaikan tugas akhir dengan judul:

“PREDIKSI CUACA UNTUK PENINGKATAN KESELAMATAN PENERBANGAN DENGAN MENGGUNAKAN JARINGAN SARAF TIRUAN DAN *PARTICLE SWARM OPTIMIZATION*”

Penulis menyadari bahwa tugas akhir ini tidak mungkin terwujud tanpa bantuan dari berbagai pihak. Perkenankan penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Agus M. Hatta, S.T. , M.Si, Ph.D selaku kepala departemen Teknik Fisika ITS.
2. Dr. Ir. Syamsul Arifin, MT. selaku dosen pembimbing tugas akhir penulis, yang selalu memberikan saran dan motivasi dalam penyelesaian tugas akhir ini.
3. Segenap Bapak/Ibu dosen pengajar di departemen Teknik Fisika - ITS.
4. Mama Evelyn Ridha Avenina Ratih dan Papa Arief Hartono Setiawan yang senantiasa memberikan do'a dan dukungan secara penuh, serta nasihat dan motivasinya.
5. Adik-adik penulis, Iwang, Itza, dan Dimas, yang telah senantiasa memberikan do'a dan dukungannya.
6. Partner tugas akhir, Dwi Andhitia Rafika, yang telah banyak membantu dan menemani selama pengerjaan tugas akhir, serta memberikan do'a dan dukungannya.
7. Segenap keluarga besar penulis yang telah memberikan do'a dan dukungannya.
8. Pihak BMKG terutama pada Bagian Ruang Observasi, dan Prakiraan Cuaca, yang telah menerima penulis dan memberikan informasi yang dibutuhkan dalam penyelesaian tugas akhir.

9. Teman-teman seperjuangan dalam mengerjakan tugas akhir yang telah memberikan banyak bantuan dalam penyelesaian tugas akhir.
10. Rekan-rekan mahasiswa Teknik Fisika - ITS, yang senantiasa memberikan motivasi.

Semoga Allah SWT senantiasa membalas segala amal dan kebaikan pihak-pihak yang telah membantu dalam proses penyelesaian tugas akhir ini. Penulis juga menyadari bahwa pada tugas akhir ini masih terdapat banyak kekurangan. Oleh karena itu, segala kritik dan saran yang membangun akan diterima dengan senang hati. Semoga tugas akhir ini dapat bermanfaat dan berguna bagi penulis, para pembaca, dan pihak-pihak yang membutuhkan.

Penulis.

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN.....	vii
ABSTRAK	xi
ABSTRACT	xiii
KATA PENGANTAR.....	xv
DAFTAR ISI.....	xvii
DAFTAR TABEL	xxi
DAFTAR GAMBAR.....	xxiii
DAFTAR NOTASI	xxv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan.....	3
1.4 Batasan Masalah.....	4
BAB II DASAR TEORI.....	5
2.1 Cuaca dan Iklim	5
2.1.1 Suhu udara	5
2.1.2 Kelembaban	6
2.1.3 Curah Hujan	6
2.1.4 Tekanan Atmosfer	7
2.1.5 Angin	8
2.1.6 Embun, Kabut, dan Awan	9
2.2 Penerapan Cuaca dan Iklim dalam Penerbangan	10
2.3 Jaringan Saraf Tiruan (JST)	11
2.3.1 Algoritma Levenberg-Marquardt	13
2.4 Particle Swarm Optimization (PSO)	15
2.4.1 Implementasi dari <i>Particle Swarm Optimization</i>	17
2.5 Standar Keselamatan Penerbangan.....	19
2.5.1 Jarak Pandang (Visibilitas)	19
2.5.2 Kecepatan dan Arah Angin	20
2.6 <i>Root Mean Square Error</i> (RMSE)	21
BAB III METODE PENELITIAN.....	23
3.1 Perumusan Masalah.....	24
3.2 Studi Literatur	24

3.3	Penentuan Variabel Penelitian	24
3.4	Pengumpulan Data Cuaca.....	24
3.5	Pengujian Data Cuaca.....	25
3.5.1	Uji Data yang Hilang	25
3.5.2	Uji Data <i>Outlier</i>	25
3.5.3	Uji Normalitas Data.....	25
3.5.4	Analisa Korelasi	25
3.5.5	Analisa Regresi	26
3.5.6	Analisa Jalur (<i>Path Analysis</i>).....	26
3.6	Perancangan Algoritma Pelatihan JST dan JST-PSO dengan <i>Software</i> MATLAB.....	26
3.6.1	Algoritma Pelatihan Jaringan Saraf Tiruan (JST)	27
3.6.2	Algoritma Pelatihan Jaringan Saraf Tiruan dengan <i>Particle Swarm Optimization</i> (JST-PSO)	28
3.7	Simulasi dan Validasi Sistem Prediksi Cuaca.....	29
3.8	Analisa Hasil.....	32
3.9	Pembahasan Hasil	33
3.10	Kesimpulan dan Saran.....	33
	BAB IV HASIL DAN PEMBAHASAN.....	35
4.1	Pengujian Data Cuaca.....	35
4.1.1	Analisa Regresi Antar Variabel Cuaca	35
4.1.2	Analisa Korelasi Antar Variabel Cuaca.....	39
4.1.3	Analisa Jalur (<i>Path Analysis</i>) Antar Variabel Cuaca.....	42
4.2	Prediksi Cuaca dengan menggunakan Jaringan Saraf Tiruan (JST)	46
4.2.1	Pelatihan dan Validasi untuk Memprediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan	46
4.2.2	Performansi Pelatihan Data Kecepatan Angin Menggunakan Jaringan Saraf Tiruan.....	49
4.2.3	Jumlah <i>Hidden Node</i> untuk Prediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan.....	50
4.2.4	Pelatihan dan Validasi untuk Memprediksi Visibilitas (Jarak Pandang) Menggunakan Jaringan Saraf Tiruan	51
4.2.5	Performansi Pelatihan Data Visibilitas Menggunakan Jaringan Saraf Tiruan.....	54

4.2.6	Jumlah <i>Hidden Node</i> untuk Prediksi Visibilitas (Jarak Pandang) Menggunakan Jaringan Saraf Tiruan	55
4.3	Prediksi Cuaca dengan menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i> (JST-PSO).....	56
4.3.1	Pelatihan dan Validasi untuk Memprediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	56
4.3.2	Performansi Pelatihan Data Kecepatan Angin Menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	59
4.3.3	Jumlah <i>Hidden Node</i> untuk Prediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	60
4.3.4	Pelatihan dan Validasi untuk Memprediksi Visibilitas Menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	61
4.3.5	Performansi Pelatihan Visibilitas Menggunakan Jaringan saraf tiruan dan <i>Particle Swarm Optimization</i>	64
4.3.6	Jumlah <i>Hidden Node</i> untuk Prediksi Visibilitas (Jarak Pandang) Menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	65
4.4	Parameter Arsitektur Jaringan Saraf Tiruan dari Hasil Percobaan.....	66
4.4.1	Fungsi Aktivasi	66
4.4.2	Faktor Bobot Optimum.....	67
4.5	Perbandingan Hasil Antara Kedua Metode	71
4.6	Keselamatan Penerbangan Menggunakan Metode Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	72
BAB V KESIMPULAN.....		75
5.1	Kesimpulan	75
5.2	Saran.....	75
DAFTAR PUSTAKA		77
LAMPIRAN		81

DAFTAR TABEL

Tabel 2.1.	Kriteria Intensitas Curah Hujan Menurut BMKG Juanda.....	7
Tabel 2.2.	Klasifikasi Tingkat Rawan Banjir Berdasarkan Curah Hujan Bulanan dan Harian Terkait Banjir Menurut BMKG Juanda	7
Tabel 4.1.	Analisa Regresi Antara Seluruh Variabel Cuaca dengan Kecepatan Angin.....	35
Tabel 4.2.	Analisa Regresi Antara Masing-Masing Variabel Cuaca dengan Kecepatan Angin	36
Tabel 4.3.	Analisa Regresi Antara Seluruh Variabel Cuaca dengan Visibilitas.....	37
Tabel 4.4.	Analisa Regresi Antara Masing-Masing Variabel Cuaca dengan Visibilitas	38
Tabel 4.5.	Koefisien Korelasi dengan Variabel Dependen Kecepatan Angin.....	40
Tabel 4.6.	Koefisien Korelasi dengan Variabel Dependen Visibilitas.....	41
Tabel 4.7.	Nilai RMSE Prediksi Kecepatan Angin dengan Menggunakan Jaringan Saraf Tiruan	48
Tabel 4.8.	Nilai RMSE Prediksi Visibilitas dengan Menggunakan Jaringan Saraf Tiruan	53
Tabel 4.9.	Nilai RMSE Prediksi Kecepatan Angin dengan Menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	58
Tabel 4.10.	Nilai RMSE Visibilitas dengan Menggunakan Jaringan Saraf Tiruan dan <i>Particle Swarm Optimization</i>	63
Tabel 4.11.	Faktor Bobot Optimum dari Metode JST untuk Prediksi Kecepatan Angin	67
Tabel 4.12.	Faktor Bobot Optimum dari Metode JST untuk Prediksi Visibilitas	68
Tabel 4.13.	Faktor Bobot Optimum dari Metode JST-PSO untuk Prediksi Kecepatan Angin	69

Tabel 4.14. Faktor Bobot Optimum dari Metode JST-PSO untuk Prediksi Visibilitas	70
Tabel 4.15. Perbandingan hasil terbaik antara metode JST dan metode gabungan JST-PSO	71
Tabel 4.16. Besar Kecepatan Angin yang Melewati Batas Aman	73

DAFTAR GAMBAR

Gambar 2.1.	Arsitektur Jaringan Saraf Tiruan	12
Gambar 2.2.	Ilustrasi <i>Particle Swarm Optimization</i> dalam mencapai target (Clerc, 2006)	15
Gambar 2.3.	Arah landasan terbang Bandar Udara Internasional Juanda, Surabaya	19
Gambar 3.1.	Diagram alir penelitian tugas akhir.....	23
Gambar 3.2.	Diagram alir algoritma jaringan saraf tiruan	27
Gambar 3.3.	Diagram alir algoritma gabungan jaringan saraf tiruan dan <i>particle swarm optimization</i>	29
Gambar 4.1.	Hipotesis awal diagram jalur antar variabel cuaca	42
Gambar 4.2.	Diagram jalur untuk prediksi kecepatan angin	44
Gambar 4.3.	Diagram jalur untuk prediksi visibilitas	45
Gambar 4.4.	Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 3 <i>hidden node</i>	47
Gambar 4.5.	Hasil validasi JST untuk data kecepatan angin dengan menggunakan 3 <i>hidden node</i>	47
Gambar 4.6.	Grafik performansi pelatihan JST untuk data kecepatan angin	49
Gambar 4.7.	Arsitektur JST untuk prediksi kecepatan angin...	51
Gambar 4.8.	Hasil pelatihan JST untuk data visibilitas dengan menggunakan 4 <i>hidden node</i>	52
Gambar 4.9.	Hasil validasi JST untuk data visibilitas dengan menggunakan 4 <i>hidden node</i>	52
Gambar 4.10.	Grafik performansi pelatihan JST untuk data visibilitas	54
Gambar 4.11.	Arsitektur JST untuk prediksi visibilitas	55
Gambar 4.12.	Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 6 <i>hidden node</i>	57
Gambar 4.13.	Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 6 <i>hidden node</i>	57
Gambar 4.14.	Grafik performansi pelatihan JST-PSO untuk data kecepatan angin	59

Gambar 4.15. Arsitektur JST-PSO untuk prediksi kecepatan angin61

Gambar 4.16. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 9 *hidden node*.....62

Gambar 4.17. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 9 *hidden node*.....62

Gambar 4.18. Grafik performansi pelatihan JST-PSO untuk data visibilitas64

Gambar 4.19. Arsitektur JST-PSO untuk prediksi visibilitas65

Gambar 4.20. Fungsi aktivasi sigmoid tangen (*binnary sigmoid*) pada JST.....66

DAFTAR NOTASI

Notasi	Nama	Satuan
TEMP	Suhu udara rata-rata harian	°F
DEWP	Titik embun rata-rata harian	°F
SLP	Tekanan udara rata-rata harian di atas permukaan laut	Milibar
STP	Tekanan udara rata-rata harian di stasiun cuaca	Milibar
MXSPD	Kecepatan angin tertinggi dalam satu hari	Knot
MAX T.	Suhu udara maksimum dalam satu hari	°F
MIN T.	Suhu udara minimum dalam satu hari	°F
VISIB	Jarak pandang atau visibilitas rata-rata harian	Mil
WDSP	Kecepatan angin rata-rata harian	Knot
z	Nilai standardisasi data	-
df	Derajat kebebasan (<i>Degree of Freedom</i>)	-
Sig.	Nilai signifikansi data	-
R	Koefisien korelasi berganda	-
R ²	Koefisien determinasi	-
B	Koefisien regresi	-
Beta	B yang telah terstandardisasi	-
v_{ij}	Bobot yang berasal dari <i>input node</i> ke- <i>i</i> ke <i>hidden node</i> ke- <i>j</i>	-
w_j	Bobot yang berasal dari <i>hidden node</i> ke- <i>j</i>	-

$S_{in\ i}$	<i>Input node</i> atau node masukan ke- <i>i</i>	-
$S_{hn\ j}$	Fungsi aktivasi pada <i>hidden node</i> atau node tersembunyi ke- <i>j</i>	-
S_{out}	Fungsi aktivasi pada <i>output node</i> atau node keluaran	-
RMSE	<i>Root Mean Square Error</i>	-

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perubahan iklim dan cuaca yang tidak menentu menjadi sebuah fenomena yang memiliki dampak buruk dan tidak dapat dihindari. Fenomena ini dapat terjadi karena beberapa faktor yaitu meningkatnya pemanasan global, jumlah karbon dioksida yang lebih banyak di atmosfer, kenaikan permukaan laut, dan sebagainya. Dampak dari perubahan iklim dan cuaca berpengaruh pada berbagai bidang, salah satunya pada bidang penerbangan.

Penerbangan merupakan salah satu sarana transportasi yang penting dan memiliki pengaruh yang sangat besar dalam berkembangnya suatu negara. Menurut data World Bank, jumlah penerbangan di Indonesia meningkat sangat pesat dari tahun 1970 hingga 2015 yaitu sebanyak 826.400 penerbangan terlaksana pada 1970 dan 88.685.767 penerbangan terlaksana pada tahun 2015 (World Bank, 2017). Dilihat dari jumlah penerbangan tersebut, kebutuhan transportasi udara sangat vital dalam berkembangnya suatu negara. Banyaknya kebutuhan transportasi udara juga harus didukung dengan teknologi yang canggih guna meningkatkan keamanan dan keselamatan para penggunanya. Salah satu teknologinya yaitu untuk melakukan prediksi cuaca sebelum penerbangan dilakukan.

Menurut data statistik dari Federal Aviation Administration (FAA), cuaca menjadi penyebab sebesar 70 % dari keseluruhan penundaan penerbangan yang terjadi. Hal tersebut akan menyebabkan kemungkinan terjadinya ketidaktepatan waktu penerbangan yang cukup besar. Cuaca juga memiliki peran yang signifikan pada kecelakaan penerbangan yaitu sebesar 23% dari seluruh penerbangan yang ada (Kulesa, 2003). Dalam upaya mengurangi kecelakaan, dibutuhkan sebuah metode yang tepat dan akurat dalam melakukan prediksi pada perubahan iklim dan cuaca, sehingga dapat membantu dalam membuat suatu keputusan pada waktu sebelum penerbangan ataupun saat penerbangan sedang berlangsung.

Di Indonesia, terdapat total 298 bandar udara dengan berbagai kelas yang terdata oleh Direktorat Jendral Perhubungan Udara dari Kementerian Perhubungan Republik Indonesia, dan terdapat 27 bandar udara internasional yang sedang beroperasi saat ini (Kementerian Perhubungan Republik Indonesia, 2017). Salah satu bandar udara internasional yang ada di Indonesia terdapat di Surabaya yaitu Bandar Udara Internasional Juanda. Pada tahun 2016, Bandar Udara Internasional Juanda dinobatkan sebagai bandar udara yang paling tepat waktu pada kategori bandar udara besar oleh perusahaan pendataan perjalanan udara, OAG, yaitu dengan performansi ketepatan waktu sebesar 90,30% (OAG Aviation Worldwide Limited, 2017). Predikat yang sangat baik tersebut harus dapat dipertahankan dengan terus mengembangkan sistem yang dapat mendukung ketepatan waktu penerbangan, salah satunya sistem prediksi cuaca.

Metode Jaringan Saraf Tiruan (JST) atau *Artificial Neural Network* (ANN) telah berkembang sangat pesat dalam beberapa tahun terakhir dan memiliki keunggulan dalam menangani permasalahan yang non-linear dan kompleks. Metode Jaringan Saraf Tiruan dengan model *Back Propagation Network* (BPN) telah banyak digunakan dan memperoleh keberhasilan pada berbagai bidang, baik digunakan untuk prediksi cuaca, prediksi polusi udara, serta berbagai bidang lainnya. Tetapi, telah ditemukan kelemahan pada metode JST bahwa metode ini menentukan secara acak kondisi awal bobot dan struktur jaringan, sehingga cukup sulit untuk menentukan nilai optimum global (Zhao, Jin, & Huang, 2010).

Particle Swarm Optimization (PSO) merupakan suatu algoritma yang digunakan untuk menentukan nilai optimum global dan telah banyak digunakan dalam bidang kecerdasan buatan dalam beberapa tahun terakhir. PSO memiliki sifat yang adaptif, optimisasi yang global, algoritma paralel, dan kemampuan dalam memecahkan masalah. Akan tetapi, PSO mempunyai kelemahan, yaitu saat algoritma mencari solusi optimal, metode ini tidak dapat menentukan solusi optimal dengan akurat. Dengan kata lain, PSO mempunyai penyesuaian yang buruk pada ruang pencarian secara lokal (Zhao H. , Jin, Huang, & Huang, 2012).

Oleh karena itu, dibutuhkan sebuah sistem prediksi cuaca untuk meningkatkan keselamatan penerbangan dengan cara menyesuaikan hal-hal yang dibutuhkan dalam penerbangan dengan keadaan cuaca yang akan terjadi menggunakan algoritma kecerdasan buatan. Untuk dapat menggunakan secara penuh optimisasi global dari PSO dan pencarian lokal yang akurat pada JST, penelitian tugas akhir ini menggabungkan metode JST dengan algoritma PSO untuk melakukan prediksi cuaca penerbangan di Bandar Udara Internasional Juanda, Surabaya.

1.2 Rumusan Masalah

Dari latar belakang diatas, maka rumusan masalah pada penelitian kali ini adalah:

- a. Apakah prediktor cuaca dengan menggunakan metode gabungan antara Jaringan Saraf Tiruan (JST) dan *Particle Swarm Optimization* (PSO) dapat meningkatkan akurasi dari prediksi cuaca?
- b. Apakah metode *Particle Swarm Optimization* (PSO) dapat menghasilkan bobot yang optimum pada Jaringan Saraf Tiruan (JST)?
- c. Apakah metode gabungan antara Jaringan Saraf Tiruan (JST) dan *Particle Swarm Optimization* (PSO) dapat melakukan prediksi cuaca untuk meningkatkan keselamatan penerbangan?

1.3 Tujuan

Adapun tujuan dari penelitian ini adalah sebagai berikut:

- a. Mengetahui apakah prediktor cuaca dengan menggunakan metode gabungan antara Jaringan Saraf Tiruan (JST) dan *Particle Swarm Optimization* (PSO) dapat meningkatkan akurasi dari prediksi cuaca.
- b. Mengetahui apakah metode *Particle Swarm Optimization* (PSO) dapat menghasilkan bobot yang optimum pada Jaringan Saraf Tiruan (JST).
- c. Mengetahui apakah metode gabungan antara Jaringan Saraf Tiruan (JST) dan *Particle Swarm Optimization* (PSO) dapat

melakukan prediksi cuaca untuk meningkatkan keselamatan penerbangan.

1.4 Batasan Masalah

Adapun batasan masalah pada tugas akhir ini adalah:

- a. Daerah yang digunakan untuk penelitian adalah pada Bandar Udara Internasional Juanda, Surabaya.
- b. Variabel yang diprediksi adalah kecepatan angin dan jarak pandang (visibilitas) rata-rata dalam satu hari.
- c. Data yang digunakan merupakan data cuaca pada Badan Meteorologi Klimatologi, dan Geofisika (BMKG) Juanda dengan rentang waktu lima tahun dari tahun 2012 hingga 2016.
- d. Peningkatan keselamatan pada penelitian ini dilakukan untuk keadaan saat pesawat mendarat (*landing*) dan lepas landas (*take off*).

BAB II

DASAR TEORI

2.1 Cuaca dan Iklim

Cuaca merupakan suatu keadaan atmosfer yang terjadi di tempat tertentu pada waktu tertentu atau pada waktu yang cukup singkat (Allaby, 2007). Penjelasan mengenai cuaca akan termasuk di dalamnya unsur-unsur cuaca yaitu suhu udara, kelembaban udara, curah hujan, tekanan atmosfer, angin, jumlah dan tipe awan, dan visibilitas atau jarak pandang. Jika dibandingkan, iklim merupakan suatu konsep yang lebih luas.

Cuaca dan iklim merupakan dua kondisi yang hampir sama, hanya berbeda dalam kurun waktunya. Iklim merupakan keadaan atmosfer dalam wilayah yang luas dan jangka waktu yang sangat panjang yaitu kurang lebih 30 tahun (Riadi & Nurmahaludin, 2012), berbeda dengan cuaca yang ditujukan pada wilayah yang sempit dan kurun waktu yang cukup singkat. Rotasi bumi menyebabkan setiap tempat mengalami perubahan cuaca dengan siklus waktu 24 jam, sedangkan revolusi bumi menyebabkan perubahan cuaca dan iklim secara teratur dan pola musim dalam jangka waktu satu tahun.

2.1.1 Suhu udara

Secara fisis suhu udara dapat didefinisikan sebagai tingkat gerakan molekul benda, semakin cepat gerakan molekul maka semakin tinggi suhunya. Suhu juga dapat didefinisikan sebagai tingkat panas suatu benda. Secara alami, panas akan bergerak dari sebuah benda yang mempunyai suhu tinggi ke benda dengan suhu yang lebih rendah.

Suhu udara dapat dinyatakan dengan berbagai skala. Dua skala yang sering dipakai dalam pengukuran udara adalah skala Fahrenheit yang dipakai di negara inggris dan skala Celcius atau skala perseratusan (*centigrade*) yang dipakai oleh sebagian besar negara di dunia.

2.1.2 Kelembaban

Kelembaban dapat direpresentasikan pada berbagai bentuk yang menunjukkan jumlah spesifik uap air yang berada di udara. Terdapat beberapa istilah untuk mendeskripsikan kelembaban, yaitu kelembaban absolut (*absolute humidity*), kelembaban spesifik (*specific humidity*), rasio pencampuran (*mixing ratio*), kelembaban relatif (*relatif humidity*), dan titik embun (*dew point*) (Ahrens, 2003). Kelembaban absolut juga dapat disebut sebagai densitas uap air yang menunjukkan massa sebuah uap air yang berada pada satuan volume udara. Kelembaban spesifik menunjukkan sebuah perbandingan antara massa uap air dengan total massa udara, dimana rasio pencampuran menunjukkan perbandingan antara massa uap air dengan total massa udara kering yang tersisa. Kelembaban relatif merupakan cara yang paling sering digunakan untuk menunjukkan tingkat kelembaban, yang mana dapat diartikan sebagai rasio perbandingan jumlah uap air aktual di udara dengan jumlah maksimum uap air yang dibutuhkan untuk saturasi pada suhu dan tekanan tertentu. Titik embun menunjukkan suhu dimana udara dapat didinginkan supaya terjadi saturasi dengan uap air atau suhu udara pada kelembaban relatif 100 persen. Semakin rendah kelembaban relatif, maka semakin rendah titik embun, yang nilainya berada dibawah suhu udara.

2.1.3 Curah Hujan

Endapan atau presipitasi didefinisikan sebagai air (H_2O) yang berbentuk cair atau padatan (es) yang jatuh ke permukaan bumi. Meskipun kabut, embun, dan embun beku (*frost*) dapat berperan dalam kelembaban (*moisture*) dari atmosfer ke permukaan bumi, unsur tersebut tidak ditinjau sebagai endapan. Bentuk endapan adalah hujan, gerimis, salju, dan hujan es (*hail*). Hujan adalah bentuk endapan yang sering dijumpai. Di Indonesia, yang dimaksud dengan endapan adalah curah hujan.

Curah hujan dan suhu merupakan unsur iklim yang sangat penting bagi kehidupan di bumi. Jumlah curah hujan dicatat dalam inci atau milimeter. Jumlah curah hujan 1 mm, menunjukkan

tinggi air hujan yang menutupi permukaan 1 mm, jika air tersebut tidak meresap ke dalam tanah atau menguap ke atmosfer.

Tabel 2.1. Kriteria Curah Hujan Menurut BMKG Juanda

No	Jenis Hujan	Intensitas Hujan (mm)	
		1 Jam	14 Jam
1	Hujan sangat ringan	< 1	< 5
2	Hujan ringan	1-5	5-20
3	Hujan normal/sedang	5-10	20-50
4	Hujan lebat	10-20	50-100
5	Hujan sangat lebat	>20	>100

Tabel 2.2. Klasifikasi Tingkat Rawan Banjir Berdasarkan Curah Hujan Bulanan dan Harian Terkait Banjir Menurut BMKG Juanda

No	Tingkat Rawan	Curah Hujan Bulanan	Curah Hujan Harian
1	Tinggi	>500 mm	>100 mm
2	Menengah/Sedang	300-500 mm	20-100 mm
3	Rendah	200-300 mm	10-20 mm

2.1.4 Tekanan Atmosfer

Berat sebuah kolom udara per satuan luas di atas sebuah titik menunjukkan tekanan atmosfer pada titik tersebut. Di permukaan laut tekanan atmosfer normal adalah 101,32 kPa atau 1.012,2 mb. Distribusi tekanan horisontal dinyatakan oleh isobar, yaitu garis yang menghubungkan tempat yang mempunyai tekanan atmosfer sama pada ketinggian tertentu. Tekanan atmosfer berubah sesuai dengan tempat dan waktu.

Karena atmosfer mengikuti hukum gas dan bersifat dapat dimampatkan (*compressible*), maka massa jenis atmosfer paling besar terdapat pada lapisan bawah karena lapisan atmosfer ini ter-tekan oleh massa atmosfer di atasnya. Tekanan atmosfer selalu berkurang dengan bertambahnya ketinggian.

2.1.5 Angin

Angin dapat didefinisikan pergerakan udara yang sejajar dengan permukaan bumi. Udara bergerak dari daerah yang mempunyai tekanan tinggi menuju ke daerah yang mempunyai tekanan rendah, pergerakan tersebut yang akan menyebabkan munculnya angin.

Angin merupakan besaran vektor yang mempunyai arah dan kecepatan (Tjasjono, 1999). Arah angin dinyatakan dalam derajat, yaitu sebagai berikut:

- a. 0° : Utara (U)
- b. $22,5^{\circ}$: Utara Timur Laut (UTL)
- c. 45° : Timur Laut (TL)
- d. $67,5^{\circ}$: Timur Timur Laut (TTL)
- e. 90° : Timut (T)
- f. $112,5^{\circ}$: Timur Tenggara (TTg)
- g. 135° : Tenggara (Tg)
- h. $157,5^{\circ}$: Selatan Tenggara (STg)
- i. 180° : Selatan (S)
- j. $202,5^{\circ}$: Selatan Barat Daya (SBD)
- k. 225° : Barat Daya (BD)
- l. $247,5^{\circ}$: Barat Barat Daya (BBD)
- m. 270° : Barat (B)
- n. $292,5^{\circ}$: Barat Barat Laut (BBL)
- o. 315° : Barat Laut (BL)
- p. $337,5^{\circ}$: Utara Barat Laut (UBL)

Arah angin selalu diartikan sebagai arah dari mana angin berhembus dan bukan kemana angin berhembus. Arah angin dapat ditunjukkan oleh alat penunjuk arah angin yaitu *wind vane* atau *aerovane*. Arah angin ditunjukkan dalam satuan derajat yang dihitung searah jarum jam dimulai dari utara yang menunjukkan 0° , tetapi biasanya dengan simbol derajat yang tidak dituliskan. Secara klimatologis, arah angin diamati dalam 8 penjuru, tetapi dalam dunia penerbangan angin diamati dalam 16 arah. Kecepatan angin dapat dinyatakan dalam satuan meter per sekon (m/s), kilometer per jam (Km/h), dan juga knot.

2.1.6 Embun, Kabut, dan Awan

Embun merupakan suatu bintik air berukuran kecil yang terbentuk akibat uap udara yang terkondensasi pada permukaan suatu objek yang memiliki suhu dibawah titik embun. Objek yang dimaksud adalah objek yang terdapat di permukaan tanah seperti ranting, daun, dan rumput. Embun biasanya terbentuk pada malam hari yang cerah dan tenang, daripada malam yang berawan dan berangin. Malam yang cerah membuat objek yang berada dekat dengan permukaan menjadi dingin secara cepat dengan melepaskan radiasi, dan malam dengan angin tenang berarti udara yang lebih dingin berada pada level permukaan bumi.

Kabut dan awan sama-sama terdiri dari tetes air yang mengumpul di udara, tetapi secara fisis terdapat perbedaan antara kabut dan awan. Kabut terbentuk di dalam udara yang dekat dengan permukaan bumi, sedangkan awan terbentuk pada lokasi yang lebih tinggi. Perbedaan mendasar antara kabut dan awan lebih ditekankan pada metode dan tempat pembentukannya. Kabut terbentuk melalui pendinginan udara oleh sentuhan dan pencampuran udara atau penjumlahan udara oleh penambahan kadar air. Kabut terjadi jika udara di dekat permukaan bumi mencapai titik embun, jika suhu naik setelah kabut terjadi maka diperkirakan kabut akan buyar. Sedangkan awan terbentuk jika udara menjadi dingin secara adiabatik melalui udara yang naik dan mengembang.

Jumlah awan merupakan sebutan untuk menunjukkan sejauh mana langit dikaburkan oleh awan. Jumlah awan dinyatakan dalam persen, tetapi lebih umum dinyatakan dalam perdelapanan dari langit yang tertutup awan. Misalnya, jumlah awan sama dengan 0 berarti langit cerah, jumlah awan sama dengan 8 berarti langit mendung.

Embun, kabut, dan awan dapat mempengaruhi jarak pandang atau visibilitas. Visibilitas merupakan suatu jarak dimana seorang pengamat dapat melihat suatu objek seperti pohon atau bangunan dengan mata telanjang. Visibilitas diukur pada stasiun cuaca dengan referensi angka pada objek yang familiar pada jarak yang diketahui dari titik observasi. Visibilitas dapat berkurang dengan

adanya tetesan air dan partikel kecil yang padat pada udara antara objek dan pengamat. Objek dapat terlihat karena terdapat cahaya yang terpantulkan dari objek menuju pengamat. Antara objek dan pengamat, beberapa cahaya terhamburkan dan beberapa terserap, sehingga hanya sebagian pantulan cahaya yang mencapai pengamat. Pada permukaan tanah, fenomena atmosferik yang dapat mengurangi visibilitas adalah kabut, asap, hembusan salju, pasir, atau debu. Sedangkan jika berada di udara, yang dapat mengurangi visibilitas adalah awan.

2.2 Penerapan Cuaca dan Iklim dalam Penerbangan

Pesawat terbang dapat terbang di dalam atmosfer bergantung pada gaya aerodinamik yang dapat menopang pesawat dan mengatur arah penerbangan. Pesawat terbang juga digerakkan oleh mesin yang membutuhkan udara dari atmosfer untuk mendukung pembakaran dan menghasilkan gaya dorong. Selanjutnya, semua sinyal untuk proses navigasi dan komunikasi juga harus dapat menembus atmosfer sebelum sebelum mencapai pesawat terbang, begitu juga untuk sinyal dari pesawat ke fasilitas pengendali di darat atau bandar udara (Mahapatra, 1999). Oleh karena itu proses penerbangan sangat dipengaruhi oleh keadaan atmosfer yaitu cuaca dan iklim.

Cuaca buruk merupakan suatu masalah yang harus diperhatikan dan menjadi masalah yang sangat serius dalam dunia penerbangan. Unsur-unsur cuaca seperti turbulensi, jumlah awan, visibilitas (jarak pandang) dan kecepatan angin merupakan faktor yang sangat menentukan dalam penerbangan (Tjasjono, 1999).

Klimatologi dalam penerbangan digunakan mulai dari penentuan lokasi bandara hingga dalam melakukan penerbangan pesawat. Lokasi bandara yang akan digunakan, baik dalam kegunaan sipil maupun militer, bergantung pada beberapa faktor seperti kecepatan angin dan jarak pandang. Pusat industri dapat mengeluarkan asap dan menghasilkan polusi yang dapat berperan dalam mengurangi jarak pandang. Dengan melakukan studi cuaca dan iklim memungkinkan untuk meminimalisir resiko dari cuaca buruk, seperti awan rendah, turbulensi, arah angin yang menyim-

pang, serta badai. Informasi cuaca yang dibutuhkan dalam penerbangan biasanya meliputi berita informasi cuaca pada saat pesawat lepas landas (*take off*), pada trayek penerbangan pesawat, pada saat pesawat melakukan pendaratan (*landing*), dan informasi cuaca pada lapangan terbang.

Fenomena atmosferik seperti cuaca dan iklim yang telah disebutkan sebelumnya memberikan dampak kepada pesawat terbang dalam berbagai aspek. Secara umum, aspek yang akan memperoleh dampaknya secara langsung yaitu sebagai berikut (Mahapatra, 1999):

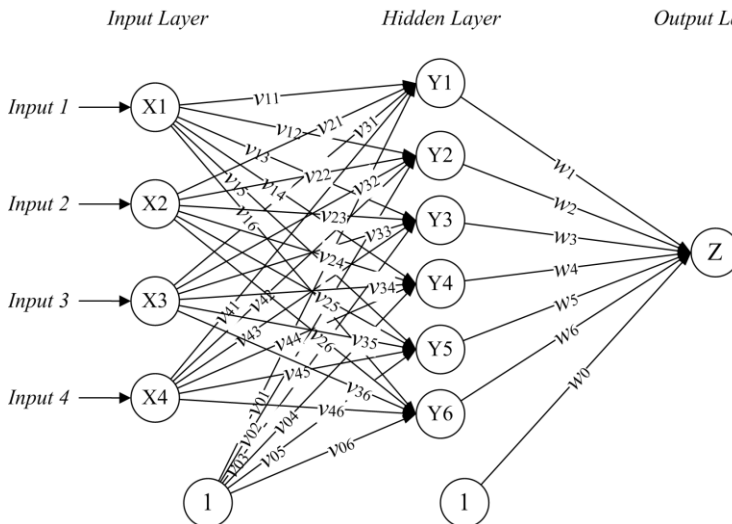
- a. Keselamatan (*Safety*)
- b. Kenyamanan (*Comfort*)
- c. Ketepatan Jadwal (*Schedule-Keeping*)
- d. Efisiensi (*Efficiency*)
- e. Ekonomi (*Economy*)
- f. Kombinasi Faktor (*Combination of Factors*)

2.3 Jaringan Saraf Tiruan (JST)

Jaringan Saraf Tiruan (JST) atau *Artificial Neural Network* (ANN) merupakan suatu sistem pemrosesan informasi yang memiliki karakteristik performansi yang mirip dengan jaringan saraf biologis. Jaringan Saraf Tiruan telah dikembangkan sebagai generalisasi dari model matematik untuk kesadaran manusia atau saraf biologis, dengan asumsi dasar sebagai berikut:

- a. Informasi yang diproses muncul sebagai beberapa elemen sederhana yang disebut neuron atau node.
- b. Sinyal dilewatkan diantara neuron menggunakan link koneksi atau koneksi yang terhubung.
- c. Setiap link koneksi membunyai bobot tertentu, dimana pada jaringan saraf secara umum, bobot tersebut akan melipatgandakan sinyal yang ditransmisikan antara neuron yang satu dengan yang lainnya.
- d. Setiap neuron mengaplikasikan fungsi aktivasi (biasanya non linear) menuju masukan jaringan (gabungan bobot dari sinyal masukan) untuk menentukan sinyal keluaran dari neuron (Fausett, 1994).

Jaringan Saraf Tiruan dikarakterisasi oleh berbagai hal yaitu, pola dari koneksi antara neuron (node) atau disebut juga dengan arsitektur jaringan, metode untuk menentukan bobot pada koneksi yang disebut juga algoritma pelatihan atau pembelajaran, dan fungsi aktivasi. Hal-hal tersebut akan memiliki pola atau nilai yang berbeda pada tiap kasus dalam menggunakan Jaringan Saraf Tiruan.



Gambar 2.1. Arsitektur Jaringan Saraf Tiruan

Sebagai contoh, gambar 2.1 merupakan contoh Jaringan Saraf Tiruan dengan menggunakan pola atau arsitektur *Multilayer Network* dengan satu layer masukan (*input layer*), satu layer tersembunyi (*hidden layer*), dan satu layer keluaran (*output layer*). Node masukan (*input node*) disimbolkan sebagai X_i node tersembunyi (*hidden node*) disimbolkan sebagai Y_j , dan node keluaran (*output node*) disimbolkan sebagai Z_k , dimana nilai i , j , dan k sesuai dengan jumlah node yang digunakan. Persamaan untuk memperoleh besar sinyal masukan ke node tersembunyi adalah sebagai berikut,

$$y_{in.j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (2.1)$$

dengan:

$y_{in,j}$ = sinyal masukan ke node tersembunyi ($j=1,2,3,\dots,n$)

v_{0j} = nilai bobot dari bias menuju node tersembunyi

x_i = sinyal keluaran dari node masukan ($i=1,2,3,\dots,m$)

v_{ij} = nilai bobot dari node masukan menuju node tersembunyi.

Persamaan untuk memperoleh besar sinyal masukan ke node keluaran adalah sebagai berikut,

$$z_{in.k} = w_{0k} + \sum_{j=1}^n y_j w_{jk} \quad (2.2)$$

dengan:

$z_{in.k}$ = sinyal masukan ke node keluaran ($k=1,2,3,\dots,p$)

w_{0k} = nilai bobot dari bias menuju node keluaran

y_j = sinyal keluaran dari node tersembunyi ($j=1,2,3,\dots,n$)

w_{jk} = nilai bobot dari node tersembunyi menuju node keluaran.

Kemudian, sinyal keluaran dari node tersembunyi dan node keluaran dapat diperoleh dengan mengaplikasikan sinyal masukan pada fungsi aktivasi. Salah satu fungsi aktivasi yang dapat digunakan yaitu bipolar sigmoid atau disebut juga tangen hiperbolik. Persamaan berikut dapat digunakan untuk mencari sinyal keluaran node tersembunyi,

$$y_j = \tanh(y_{in,j}) \quad (2.3)$$

dengan:

y_j = sinyal keluaran dari node tersembunyi ($j=1,2,3,\dots,n$).

Persamaan mencari sinyal keluaran dari node keluaran adalah sebagai berikut,

$$z_k = \tanh(z_{in.k}) \quad (2.4)$$

dengan:

z_k = sinyal keluaran dari node keluaran ($k=1,2,3,\dots,p$).

2.3.1 Algoritma Levenberg-Marquardt

Algoritma Levenberg-Marquardt merupakan salah satu jenis dari algoritma pelatihan JST *Backpropagation* dengan dua jenis perhitungan, yaitu perhitungan maju dan perhitungan mundur. Langkah dasar algoritma Levenberg-Marquardt adalah penentuan matriks Hessian untuk mencari bobot-bobot dan bias koneksi yang digunakan. Matriks Hessian merupakan turunan kedua dari fungsi kinerja terhadap masing-masing komponen bobot dan bias.

Matriks Hessian diubah dengan pendekatan secara iteratif pada masing-masing *epoch* selama algoritma pelatihan berjalan dengan tujuan untuk memudahkan komputasi. *Epoch* pada jaringan saraf tiruan merupakan sebuah proses pengulangan untuk melakukan pembaharuan bobot dengan menggunakan seluruh data pelatihan. Secara singkat perhitungan Levenberg-Marquardt dapat dijelaskan sebagai berikut:

- a. Inisialisasi bobot awal secara acak, lalu hitung total *error*.
- b. Lakukan pembaharuan bobot secara langsung dengan rumus berikut.

$$w_{k+1} = w_k - (J_k^T J_k + \mu I)^{-1} J_k e_k \quad (2.5)$$

dengan:

w_{k+1} = bobot baru

w_k = bobot saat ini

J = matriks jacobian

μ = koefisien kombinasi (bernilai positif)

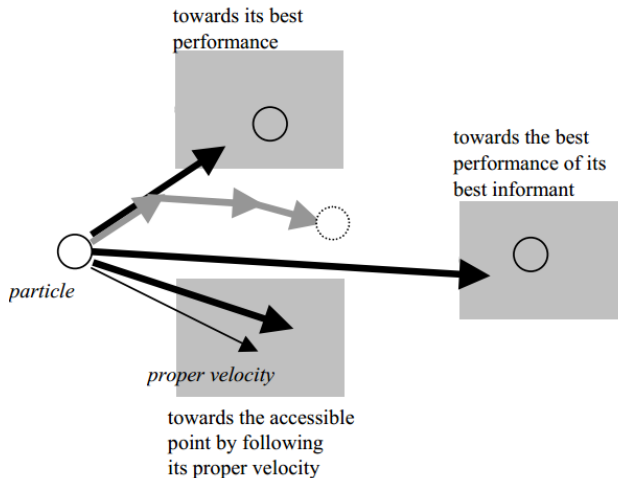
I = matriks identitas

e_k = *error* saat ini.

- c. Evaluasi nilai *error* keseluruhan dengan menggunakan bobot baru yang telah diperoleh.
- d. Jika pembaharuan bobot menyebabkan nilai *error* meningkat, maka kembali ke langkah b dengan mengubah nilai koefisien kombinasi menjadi lebih besar dengan faktor 10 atau faktor lainnya. Lalu kembali melakukan pembaharuan nilai *error* menggunakan bobot yang baru.
- e. Jika pembaharuan bobot menyebabkan nilai *error* berkurang, maka nilai bobot tersebut dapat diterima. Simpan vektor bobot yang baru sebagai bobot yang sekarang dan turunkan nilai koefisien kombinasi dengan faktor 10 atau faktor lainnya.
- f. Ulangi kembali langkah b sampai dengan e, sehingga total *error* saat ini menjadi lebih kecil dari *error* tujuannya atau hingga kondisi yang diinginkan sudah terpenuhi.

2.4 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) didasari oleh perilaku koloni atau kawanan serangga, seperti semut, rayap, lebah, atau tawon, kawanan burung, ataupun kawanan ikan. Algoritma PSO mengikuti perilaku dari organisme sosial tersebut. Kata *particle* atau partikel mengacu pada lebah di koloni, atau burung di kawanan. Setiap individu atau partikel pada suatu kawanan memiliki jalan yang tersebar menggunakan kecerdasan masing-masing dan kecerdasan kolektif atau grup dari sebuah kawanan. Sebagai contoh, jika satu partikel menemukan jalan menuju makanan, sisa kawanan yang lain akan dapat langsung mengikuti jalan tersebut, walaupun sedang berada di tempat yang jauh dalam kawanan. Metode optimisasi yang berdasarkan kecerdasan kawanan disebut algoritma yang terinspirasi dari perilaku, berbeda dengan algoritma genetik, yang biasa disebut algoritma yang berdasarkan evolusi.



Gambar 2.2. Ilustrasi *Particle Swarm Optimization* dalam mencapai target (Clerc, 2006)

Metode optimisasi dengan metode *particle swarm optimization* bertujuan untuk mencari nilai optimum dari suatu fungsi

objektif. Nilai optimum yang dimaksud dapat berupa nilai minimum atau nilai maksimum sesuai dengan fungsi objektifnya. Dalam konsep optimisasi, suatu kawanan diasumsikan dengan ukuran yang spesifik dengan tiap partikel awalnya terletak secara acak pada ruang multidimensi. Setiap partikel bergerak pada ruang tersebut dan mengingat posisi terbaik (dalam hal mencari makanan atau nilai fungsi objektif) yang telah didapatkan. Partikel akan membagikan informasi antara yang satu dengan yang lain, dan menyesuaikan posisi dan kecepatan berdasarkan informasi yang telah diterima.

Sebagai contoh, diibaratkan sebagai sekawanan burung. Walaupun masing-masing burung memiliki kecerdasan yang terbatas, mereka akan mengikuti aturan sederhana sebagai berikut:

- a. Mereka akan mencoba untuk tidak terlalu dekat dengan burung lain.
- b. Mereka akan bergerak menuju ke arah rata-rata yang dituju burung lainnya.
- c. Mereka akan mencoba untuk menyesuaikan posisi rata-rata antara burung yang lain dengan jarak yang tidak terlalu jauh dalam kawanan.

Perilaku dari kawanan burung tersebut merupakan kombinasi antara tiga faktor sederhana, yaitu *cohesion* (bergerak bersama-sama), *separation* (tidak bergerak terlalu dekat), dan *alignment* (mengikuti arah gerak secara umum dari kawanan). PSO dikembangkan berdasarkan model sebagai berikut :

- a. Saat satu burung menemukan target atau makanan atau nilai maksimum dari suatu fungsi objektif, burung itu akan secara langsung menyampaikan informasi kepada burung lain.
- b. Semua burung akan bergerak menuju target atau makanan atau nilai maksimum dari suatu fungsi objektif, tetapi tidak secara langsung.
- c. Terdapat komponen pada pemikiran setiap burung yang akan mengingat posisi sebelumnya.

Model tersebut akan menstimulasi pencarian acak dengan desain ruang untuk nilai maksimum dari fungsi objektif. Akan terjadi banyak pengulangan iterasi, hingga akhirnya burung atau

partikel tersebut mencapai target atau nilai maksimum dari fungsi objektif (Rao, 2009). Iterasi pada *particle swarm optimization* merupakan sebuah pengulangan pada saat algoritma melakukan proses untuk memperoleh solusi yang baru.

2.4.1 Implementasi dari *Particle Swarm Optimization*

Terdapat sebuah fungsi yang dinyatakan dalam persamaan sebagai berikut:

$$X^{(B)} \leq X \leq X^{(A)} \quad (2.6)$$

dimana $X^{(B)}$ adalah batas bawah dan $X^{(A)}$ adalah batas atas dari X . Langkah-langkah yang digunakan untuk mencari nilai $f(x)$ minimal dengan metode PSO yaitu sebagai berikut:

- a. Asumsikan bahwa ukuran kelompok atau jumlah partikel adalah N . Ukuran N sebaiknya tidak terlalu besar tetapi juga tidak terlalu kecil untuk mengurangi jumlah evaluasi fungsi yang diperlukan untuk menemukan solusi dengan tetap memiliki banyak kemungkinan posisi menuju solusi terbaik atau optimal. Jika terlalu kecil, sedikit kemungkinan menemukan posisi partikel yang baik, sedangkan jika terlalu besar juga akan membuat perhitungan jadi panjang. Biasanya digunakan ukuran partikel adalah 20 sampai 30 partikel.
- b. Bangkitkan populasi awal X dengan rentang $X^{(B)}$ dan $X^{(A)}$ secara acak sehingga didapat X_1, X_2, \dots, X_N . setelah itu, untuk mudahnya partikel j dan kecepatan pada iterasi I dinotasikan sebagai $X_j^{(i)}$ dan $V_j^{(i)}$. Sehingga partikel-partikel awal ini akan menjadi $X_1(0), X_2(0), \dots, X_N(0)$. Vektor $X_j(0)$, ($j = 1, 2, \dots, N$) disebut partikel atau vektor koordinat dari partikel. Evaluasi nilai fungsi tujuan untuk setiap dan nyatakan dengan,

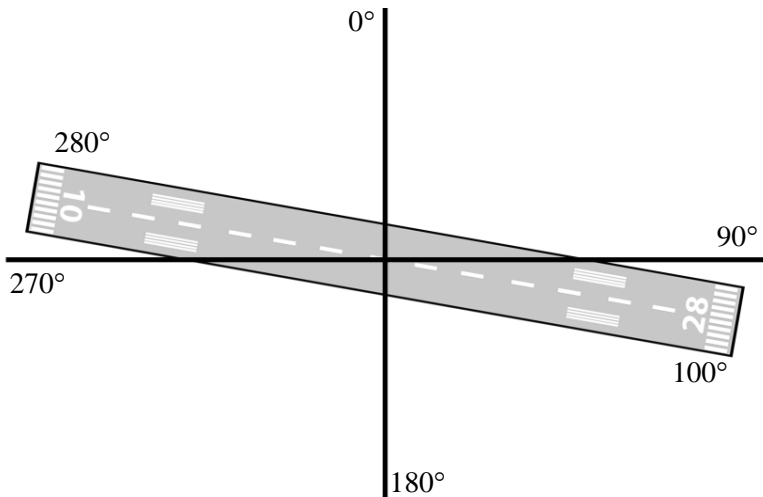
$$f|X_1(0)|, f|X_2(0)|, \dots, f|X_N(0)| \quad (2.7)$$
- c. Hitung kecepatan dari semua partikel. Semua partikel bergerak menuju titik optimal dengan suatu kecepatan. Awalnya semua kecepatan dari partikel diasumsikan dengan nol. Atur nilai iterasi $i = 1$.

- d. Pada iterasi ke- i , temukan parameter-parameter penting untuk setiap partikel j yaitu:
- Nilai terbaik sejauh ini dari $X_j(i)$, (koordinat partikel j pada iterasi i) dan nyatakan sebagai $P_{best,j}$ dengan nilai fungsi obyektif paling rendah untuk kasus minimasi, $f|X_j(i)|$ yang ditemui sebuah partikel j pada semua ite-rasi sebelumnya. Nilai terbaik untuk semua partikel $X_j(i)$ yang ditemukan sampai iterasi ke- I , G_{best} dengan nilai fungsi tujuan paling kecil atau minimum diantara semua partikel untuk semua iterasi sebelumnya, $f|X_j(i)|$
 - Hitung kecepatan partikel j pada iterasi ke i dengan rumus sebagai berikut,
- $$V_j(i) = V_j(i-1) + c_1 r_1 [P_{best,j} - x_j(i-1)] + c_2 r_2 [G_{best} - x(i-1)]; j = 1, 2, \dots, N \quad (2.8)$$
- Dimana c_1 dan c_2 masing-masing adalah *learing rates* untuk kemampuan individu (*cognitive*) dan pengaruh sosial (*group*), dan r_1 dan r_2 bilangan random yang berdistribusi uniform dalam interval 0 dan 1. Jadi, para-meter c_1 dan c_2 biasanya adalah 2, sehingga perkalian $c_1 r_1$ dan $c_2 r_2$ akan memastikan bahwa partikel-partikel akan mendekati target sekitar setengah selisihnya.
- Hitung posisi atau koordinat partikel j pada iterasi ke- I dengan cara,
- $$X_j(i) = X_j(i-1) + V_j(i); j = 1, 2, \dots, N \quad (2.9)$$
- Evaluasi nilai fungsi tujuan untuk setiap partikel dan nyatakan sebagai,
- $$f|X_1(i)|, f|X_2(i)|, \dots, f|X_N(i)| \quad (2.10)$$
- e. Cek apakah solusi yang sekarang sudah konvergen. Jika posisi semua partikel menuju ke satu nilai yang sama maka ini disebut konvergen. Jika belum konvergen maka langkah d diulangi dengan memperbarui iterasi $i=i+1$, dengan cara menghitung nilai baru dari $P_{best,j}$ dan G_{best} . Proses iterasi ini dilanjutkan sampai semua partikel menuju ke satu titik solusi yang sama. Biasanya akan ditentukan dengan kriteria

penghentian misalnya jumlah selisih saat ini dengan solusi sebelumnya sudah sangat kecil.

2.5 Standar Keselamatan Penerbangan

Setiap bandar udara memiliki standar keselamatan untuk pendaratan (*landing*) dan lepas landas (*take-off*) yang berbeda-beda tergantung dari kondisi dari lapangan terbang. Bahkan dalam satu landasan, jika pesawat bergerak pada arah yang berbeda, maka akan memiliki standar keselamatan tersendiri pada masing-masing arah pergerakan pesawat tergantung dari sarana yang terdapat pada landasan terbang. Hal ini dilakukan untuk meningkatkan efisiensi dari pendaratan dan lepas landas dengan tetap mempertahankan standar keselamatan yang tinggi.



Gambar 2.3. Arah landasan terbang Bandar Udara Internasional Juanda, Surabaya

Bandar Udara Internasional Juanda memiliki satu buah landasan terbang yang melintang dari timur ke barat dengan kemiringan sebesar 10° seperti pada gambar 2.3. Pada landasan tersebut, pesawat dapat melakukan pendaratan ataupun melakukan le-

pas landas dari arah barat ke timur dengan menggunakan landasan terbang barat (RWY 10) dan dari arah timur ke barat dengan menggunakan landasan terbang timur (RWY 28).

2.5.1 Jarak Pandang (Visibilitas)

Pilot harus dapat melihat dengan jelas posisi landasan pada saat mengemudikan pesawat untuk melakukan pendaratan supaya tidak salah dalam melakukan pendaratan. Pesawat harus dapat mendarat di landasan terbang (*runway*). Jika ingin mendarat di tempat selain landasan terbang, seperti di *taxi way*, akan diperbolehkan hanya jika pesawat dalam keadaan darurat dan dengan seijin *Air Traffic Controller* (ATC).

Pesawat dapat mendarat di Bandar Udara Internasional Juanda pada dua arah yang berbeda yaitu dari arah barat dan dari arah timur. Jarak pandang minimal saat mendarat pada Bandar Udara Internasional Juanda, Surabaya adalah sebagai berikut:

a. *Runway* Barat (RWY 10)

Karena dilengkapi dengan sarana ILS (*Instrumented Landing System*), maka jarak penglihatan minimal saat melakukan pendaratan adalah sejauh 800 meter.

b. *Runway* Timur (RWY 28)

Karena tidak dilengkapi dengan sarana ILS (*Instrumented Landing System*), maka jarak penglihatan minimal saat melakukan pendaratan adalah sejauh 1600 meter. (BMKG Juanda, 2013)

2.5.2 Kecepatan dan Arah Angin

Penentuan arah landasan terbang pada bandar udara dilakukan dengan studi arah dan kecepatan angin terlebih dahulu. Arah landasan terbang ditentukan dengan melihat arah angin rata-rata pada setiap bulannya. Hal ini dilakukan karena pada saat pesawat melakukan pendaratan ataupun lepas landas harus melawan arah darimana angin berasal.

Pada Bandar Udara Internasional Juanda, arah angin rata-rata bulanan bertiup dari arah timur ke barat, dan juga sebaliknya. Oleh karena itu, landasan terbang pada Bandar Udara Interna-

sional Juanda dibuat melintang dari arah timur ke barat dengan kemiringan sebesar 10° .

Batas keselamatan kecepatan angin untuk melakukan pendaratan ataupun lepas landas yaitu tiupan angin lebih dari 10 knot dengan arah tegak lurus (sekitar 0° dan 180°) dari arah kanan ataupun kiri landasan. Kondisi tersebut dapat dianggap bahwa landasan terbang tidak aman untuk melakukan pendaratan ataupun lepas landas (BMKG Juanda, 2013).

2.6 Root Mean Square Error (RMSE)

Evaluasi performansi model Jaringan Saraf Tiruan (JST) antara nilai aktual dan nilai yang diprediksi dapat dilakukan dengan menggunakan metode untuk mengukur nilai kesalahan atau nilai error. Metode pengukuran yang banyak digunakan salah satunya adalah *Root Mean Squared Error* (RMSE) (Ghaffari, Abdollahi, Khoshayand, Bozchalooi, Dadgar, & Rafiee-Tehrani, 2008). *Root Mean Squared Error* juga dapat dilakukan untuk mengukur akurasi dan *reliability* untuk kalibrasi dan tes data set (Keshavarzi & Sarmadian, 2010). Berikut ini adalah rumus dari RMSE:

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (Y' - Y)^2}{n}} \quad (2.11)$$

dengan:

Y = data aktual

Y' = data hasil prediksi

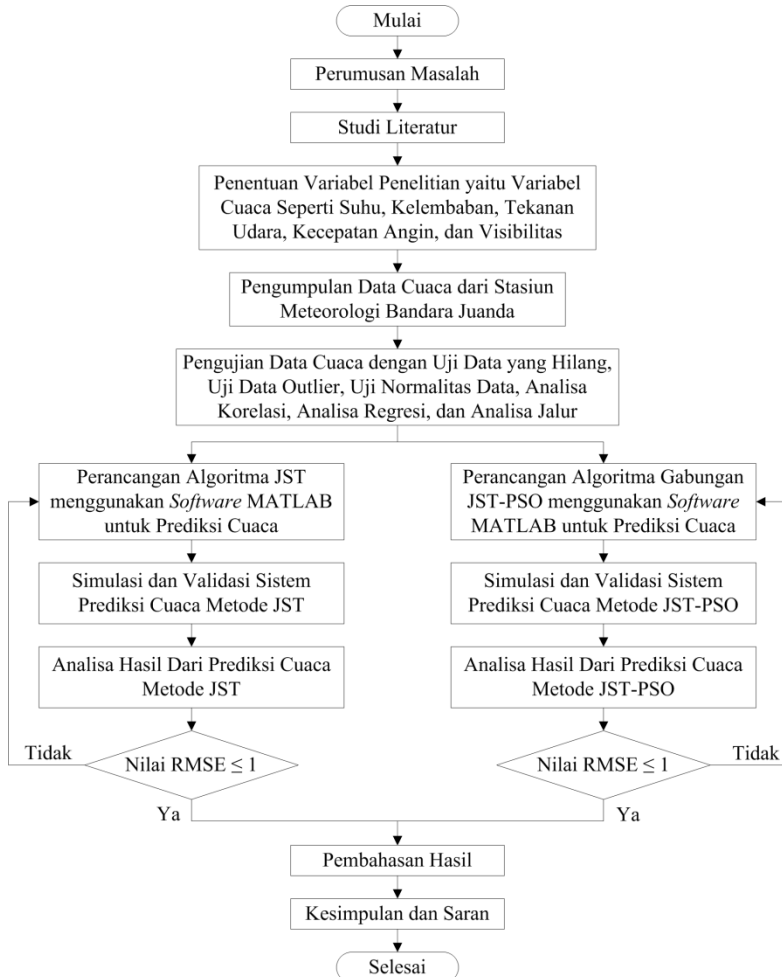
n = jumlah data.

Halaman ini sengaja dikosongkan

BAB III

METODE PENELITIAN

Berikut ini merupakan diagram alir dari pengerjaan penelitian Tugas Akhir yang telah dilakukan.



Gambar 3.1. Diagram alir penelitian tugas akhir

3.1 Perumusan Masalah

Perumusan masalah merupakan suatu masalah yang akan diselesaikan dengan metode pada penelitian tugas akhir ini dan juga menjadi pedoman dan fokus untuk mengerjakan penelitian tugas akhir.

3.2 Studi Literatur

Studi literatur merupakan kegiatan yang dilakukan untuk mempelajari prinsip dasar dalam melakukan penelitian ini, yaitu variabel untuk memprediksi cuaca, metode jaringan saraf tiruan (JST) dan *particle swarm optimization* (PSO), dan batasan yang harus dipenuhi untuk keselamatan penerbangan. Literatur yang digunakan adalah buku, jurnal penelitian ilmiah, artikel ilmiah, laporan tugas akhir, dan juga informasi yang diperoleh dari pihak Bandar Udara Internasional Juanda, Surabaya.

3.3 Penentuan Variabel Penelitian

Variabel penelitian yang ditentukan adalah variabel cuaca yang akan diprediksi atau menjadi data keluaran, dan variabel apa yang akan digunakan sebagai data masukan untuk melakukan prediksi. Informasi mengenai hal ini dapat diperoleh dari jurnal penelitian yang berhubungan dengan prediksi cuaca, kemudian dianalisa kembali untuk melihat hubungan antara variabel-variabel cuaca tersebut.

3.4 Pengumpulan Data Cuaca

Pada penelitian ini, data yang dikumpulkan adalah data laporan cuaca penerbangan pada Badan Meteorologi, Klimatologi, dan Geofisika untuk wilayah Bandar Udara Internasional Juanda yang diperoleh dari National Climatic Data Center, U.S. Department of Commerce. Data yang diambil adalah data tahun 2012-2016 pada wilayah tersebut. Beberapa variabel data laporan cuaca yang dibutuhkan adalah data suhu, kelembaban, tekanan udara, visibilitas, dan kecepatan angin. Dari data yang telah diperoleh tersebut kemudian proses prediksi cuaca dapat dilakukan.

3.5 Pengujian Data Cuaca

Dari data yang telah diambil, kemudian dilakukan uji dan analisa data cuaca. Hal ini dilakukan untuk menemukan variabel cuaca apa saja yang tepat sebagai variabel masukan untuk melakukan prediksi yang akan dilakukan. Uji dan analisa data cuaca yang dilakukan menggunakan *software* IBM SPSS Statistics 24, yaitu sebagai berikut:

3.5.1 Uji Data yang Hilang

Pengujian data yang hilang atau *missing data* dilakukan untuk menguji data yang tidak lengkap atau adanya data hilang yang akan mempengaruhi pengolahan data secara keseluruhan. Dalam penggunaan SPSS, *missing data* adalah adanya sel yang kosong pada satu atau beberapa variabel. Data yang hilang terjadi karena informasi untuk suatu objek tidak diberikan, sulit dicari, atau memang tidak ada informasi.

3.5.2 Uji Data *Outlier*

Data *outlier* adalah data yang secara nyata berbeda dengan data-data yang lainnya, atau dengan kata lain memiliki nilai yang jauh lebih besar ataupun jauh lebih kecil dibanding data lainnya. Keberadaan data *outlier* atau data yang sangat ekstrem, pada banyak kasus akan mengganggu keseluruhan data yang dapat menyebabkan biasnya kesimpulan yang diambil.

3.5.3 Uji Normalitas Data

Pengujian normalitas data dilakukan untuk mengetahui apakah sebuah data mengikuti atau mendekati distribusi normal, yaitu distribusi data dengan bentuk lonceng. Data yang baik adalah data yang mempunyai pola distribusi normal.

3.5.4 Analisa Korelasi

Analisa korelasi dilakukan untuk mencari hubungan antara dua variabel atau lebih dengan cara menghitung koefisien korelasi antar variabel yang akan dicari hubungannya. Korelasi merupakan angka yang menunjukkan arah dan kuatnya hubungan antar

dua variabel atau lebih. Arah dinyatakan dengan nilai positif atau negatif, sedangkan kuatnya hubungan dinyatakan dengan besarnya koefisien korelasi. Koefisien korelasi positif terbesar adalah 1, koefisien korelasi negatif terbesar adalah -1, sedangkan koefisien korelasi terkecil adalah 0. Semakin kecil nilai koefisien korelasi, maka akan semakin besar error untuk membuat prediksi.

3.5.5 Analisa Regresi

Analisa regresi digunakan untuk memprediksikan seberapa jauh perubahan nilai variabel dependen (terikat), bila nilai variabel independen (bebas) dimanipulasi atau diubah. Bila nilai koefisien korelasi tinggi, pada umumnya koefisien regresi juga tinggi, sehingga daya prediktifnya akan tinggi. Manfaat dari hasil analisa regresi adalah untuk membuat keputusan apakah naik dan turunnya variabel dependen dapat dilakukan melalui peningkatan variabel independen atau tidak.

3.5.6 Analisa Jalur (*Path Analysis*)

Analisa jalur digunakan untuk melukiskan dan menguji model hubungan antar variabel yang berbentuk sebab akibat, bukan berbentuk hubungan interaktif atau *reciprocal*.

Dalam melakukan analisa jalur, terdapat tiga pertanyaan yang perlu diperhatikan (Trihendradi, 2012), yaitu:

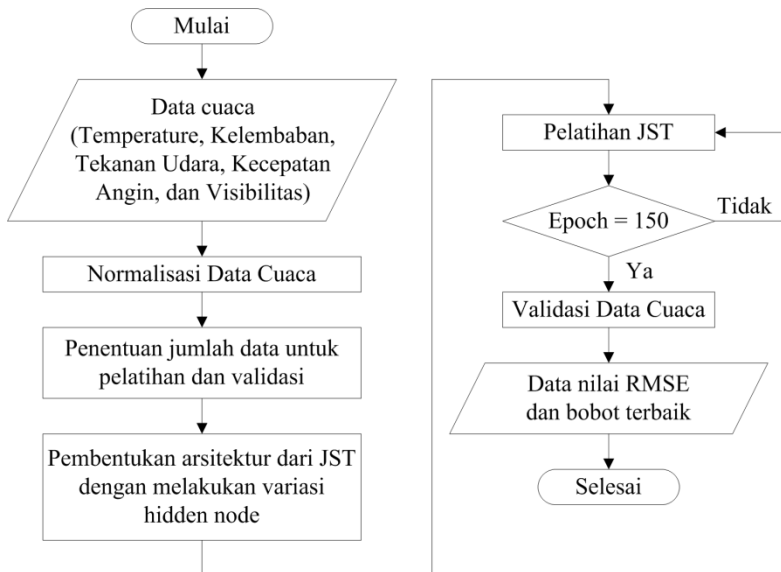
- a. Apa saja variabel kausal yang terlibat dalam diagram?
- b. Bagaimana urutan rantai kausal pada variabel tersebut?
- c. Jalur mana yang tidak penting dalam diagram setelah dilakukan analisa regresi dan korelasi?

3.6 Perancangan Algoritma Pelatihan JST dan JST-PSO dengan *Software* MATLAB

Terdapat dua jenis algoritma yang digunakan untuk melakukan prediksi cuaca yaitu dengan menggunakan algoritma jaringan saraf tiruan (JST) dan dengan menggunakan algoritma gabungan antara metode jaringan saraf tiruan dengan *particle swarm optimization* (JST-PSO). Jaringan saraf tiruan disini merupakan algoritma utama, sedangkan *particle swarm optimization*

digunakan untuk menentukan bobot optimal yang akan digunakan pada jaringan saraf tiruan. Kemudian kedua algoritma tersebut akan dibandingkan hasil akhirnya untuk mengetahui algoritma yang mana yang memiliki akurasi atau ketepatan prediksi yang lebih baik.

3.6.1 Algoritma Pelatihan Jaringan Saraf Tiruan (JST)



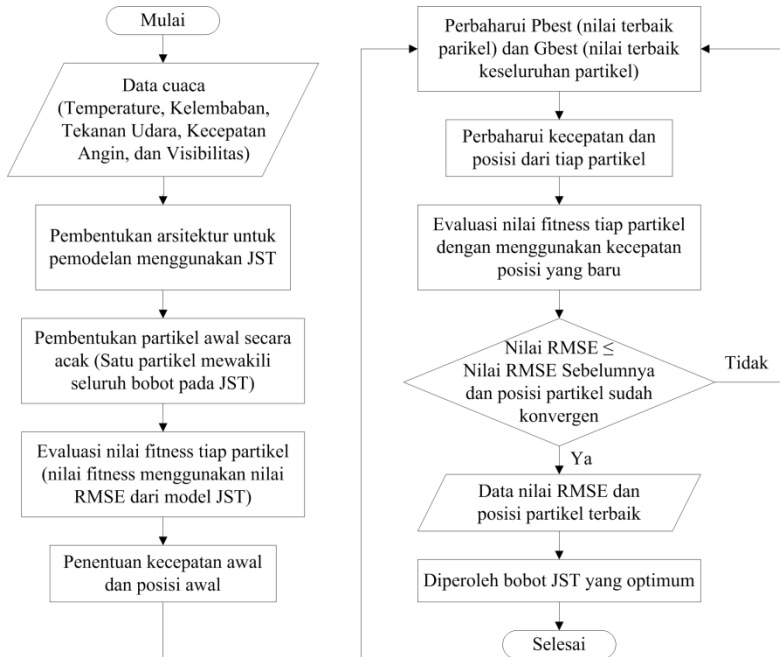
Gambar 3.2. Diagram alir algoritma jaringan saraf tiruan

Algoritma pertama yang digunakan untuk melakukan prediksi cuaca adalah algoritma yang hanya menggunakan model jaringan saraf tiruan. Langkah pertama yaitu mengambil data cuaca sesuai dengan variabel cuaca yang dibutuhkan dalam melakukan prediksi kecepatan angin dan visibilitas. Data tersebut kemudian dilakukan normalisasi sehingga data dari seluruh variabel cuaca memiliki *range* yang sama yaitu 0 hingga 1. Setelah itu, jumlah data untuk pelatihan dan validasi ditentukan. Jumlah data yang digunakan adalah sebanyak 1821 set data cuaca, dengan 1365 set

data digunakan untuk pelatihan, dan 456 set data digunakan untuk validasi prediksi. Kemudian dilakukan percobaan dengan melakukan variasi pada arsitekturnya yaitu variasi jumlah *hidden node* yaitu dimulai dari 1 hingga 10 *hidden node*. Kemudian dilakukan pelatihan data menggunakan algoritma pembelajaran dari jaringan saraf tiruan hingga *epoch* maksimal yaitu 150 *epoch*. Hasil dari data pelatihan tersebut kemudian divalidasi dan dilihat nilai RMSE yang terbaik, sehingga akan diperoleh jumlah *hidden node* yang tepat untuk melakukan prediksi cuaca.

3.6.2 Algoritma Pelatihan Jaringan Saraf Tiruan dengan *Particle Swarm Optimization* (JST-PSO)

Algoritma kedua yang digunakan untuk melakukan prediksi cuaca adalah algoritma gabungan antara model jaringan saraf tiruan dengan *particle swarm optimization*. Dapat dilihat pada diagram alir, data cuaca diambil sesuai dengan variabel cuaca yang dibutuhkan dalam melakukan prediksi kecepatan angin dan visibilitas. Arsitektur jaringan saraf tiruan akan digunakan sebagai model untuk mencari nilai *fitness*-nya yaitu dengan menggunakan nilai RMSE. Partikel awal diinisialisasikan secara acak, yang mana satu partikel akan mewakili seluruh jumlah bobot yang terdapat pada arsitektur jaringan saraf tiruan. Kemudian, evaluasi nilai *fitness* menggunakan tiap partikel awal. Nilai *fitness* diperoleh menggunakan fungsi objektif persamaan RMSE yang bersumber dari model jaringan saraf tiruan. Tentukan kecepatan awal dan posisi awal dari partikel, lalu perbaharui Pbest (nilai terbaik masing-masing partikel) dan G best (nilai terbaik dari keseluruhan partikel). Dari nilai tersebut kemudian perbaharui kecepatan dan posisi setiap partikel. Kemudian evaluasi nilai *fitness* dengan menggunakan posisi dan kecepatan yang baru dari setiap partikel. Lalu evaluasi nilai *fitness* apakah sudah konvergen dan posisi partikel saling mendekati. Jika belum, ulangi dari perbaharui Pbest dan Gbest hingga evaluasi *fitness*. Jika sudah, maka simpan hasilnya yaitu berupa nilai RMSE serta posisi partikel yang terbaik yang kemudian akan dijadikan bobot untuk model jaringan saraf tiruan.



Gambar 3.3. Diagram alir algoritma gabungan jaringan saraf tiruan dan *particle swarm optimization*

Algoritma pelatihan dari metode gabungan jaringan saraf tiruan dan *particle swarm optimization* dapat dituliskan secara matematis sebagai berikut:

- Tentukan arsitektur jaringan saraf tiruan yang akan digunakan atau divariasikan sebagai model untuk prediksi, yang meliputi jumlah node masukan, node tersembunyi, dan node keluaran.
- Tentukan jumlah partikel yang akan digunakan. Satu partikel merepresentasikan seluruh bobot yang ada pada jaringan saraf tiruan. Jumlah partikel yang digunakan adalah 40 partikel.
- Bangkitkan populasi awal yang berisi 40 partikel. Pada populasi awal tersebut berisi informasi mengenai posisi

koordinat awal partikel ($X_j^{(i)}$) dan kecepatan awal partikel ($V_j^{(i)}$).

- d. Evaluasi fungsi objektif untuk setiap partikel $f(X_j)$. Fungsi objektif yang digunakan adalah RMSE dengan rumus sebagai berikut:

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (z_k - z)^2}{n}} \quad (3.1)$$

dengan:

z = data aktual

z_k = data hasil prediksi

n = jumlah data

Kemudian untuk memperoleh nilai keluaran dari jaringan saraf tiruan dibutuhkan nilai sinyal masukan ke node tersembunyi sebagai berikut,

$$y_{in.j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (3.2)$$

dengan:

$y_{in.j}$ = sinyal masukan ke node tersembunyi ($j=1,2,3,\dots,n$)

v_{0j} = nilai bobot dari bias menuju node tersembunyi

x_i = sinyal keluaran dari node masukan ($i=1,2,3,\dots,m$)

v_{ij} = nilai bobot dari node masukan menuju node tersembunyi,

dan memperoleh nilai sinyal masukan ke node keluaran sebagai berikut,

$$z_{in.k} = w_{0k} + \sum_{j=1}^n y_j w_{jk} \quad (3.3)$$

dengan:

$z_{in.k}$ = sinyal masukan ke node keluaran ($k=1,2,3,\dots,p$)

w_{0k} = nilai bobot dari bias menuju node keluaran

y_j = sinyal keluaran dari node tersembunyi ($j=1,2,3,\dots,n$)

w_{jk} = nilai bobot dari node tersembunyi menuju node keluaran.

Kemudian untuk memperoleh sinyal keluaran dari node tersembunyi dan node keluaran dapat diperoleh dengan mengaplikasikan sinyal masukan pada fungsi aktivasi. Salah satu fungsi aktivasi yang dapat digunakan yaitu bipolar

sigmoid atau disebut juga tangen hiperbolik. Sehingga, dapat digunakan persamaan berikut untuk mencari sinyal keluaran node tersembunyi,

$$y_j = \tanh(y_{in,j}) \quad (3.4)$$

dengan:

y_j = sinyal keluaran dari node tersembunyi ($j=1,2,3,...,n$),

dan persamaan berikut untuk mencari sinyal keluaran dari node keluaran atau hasil prediksi jaringan saraf tiruan yaitu sebagai berikut,

$$z_k = \tanh(z_{in,k}) \quad (3.5)$$

dengan:

z_k = sinyal keluaran dari node keluaran ($k=1,2,3,...,p$).

- e. Hitung kecepatan dari semua partikel. Semua partikel bergerak menuju titik optimal dengan suatu kecepatan. Awalnya semua kecepatan dari partikel diasumsikan dengan nol.
- f. Pada setiap iterasi, temukan parameter-parameter penting untuk setiap partikel yaitu:
 - Nilai terbaik sejauh ini dari koordinat partikel dan nyatakan sebagai $P_{best,j}$ dengan nilai fungsi obyektif paling rendah untuk sebuah partikel j pada semua iterasi sebelumnya. Nilai terbaik untuk semua partikel yang ditemukan sampai iterasi tersebut atau G_{best} dengan nilai fungsi tujuan paling kecil atau minimum diantara semua partikel untuk semua iterasi sebelumnya.
 - Hitung kecepatan partikel pada iterasi tersebut dengan rumus sebagai berikut,

$$V_j(i) = V_j(i-1) + c_1 r_1 [P_{best,j} - x_j(i-1)] + c_2 r_2 [G_{best} - x(i-1)]; j = 1, 2, \dots, N \quad (3.6)$$

Dimana c_1 dan c_2 masing-masing adalah *learning rates* untuk kemampuan individu dan pengaruh sosial, dan r_1 dan r_2 bilangan random yang berdistribusi uniform dalam interval 0 dan 1.

- Hitung posisi atau koordinat partikel j pada iterasi ke- i dengan cara,

$$X_j(i) = X_j(i-1) + V_j(i); j = 1, 2, \dots, N \quad (3.7)$$

Evaluasi nilai fungsi tujuan untuk setiap partikel dan nyatakan sebagai bobot terbaik.

- g. Cek apakah solusi yang sekarang sudah konvergen. Jika posisi semua partikel menuju ke satu nilai yang sama maka ini disebut konvergen. Jika belum konvergen maka langkah d hingga g diulangi. Proses iterasi ini dilanjutkan sampai semua partikel menuju ke satu titik solusi yang sama. Selain itu, biasanya pada algoritma ini akan ditentukan kriteria penghentian.
- h. Salah satu kriteria penghentian yaitu apabila sudah menemukan bobot optimum. Faktor bobot dikatakan sudah optimum apabila nilai RMSE sudah mendekati nilai nol atau sudah memiliki nilai kurang dari satu. Waktu dalam menjalankan algoritma juga perlu diperhatikan, jangan sampai algoritma berjalan terlalu lama.

3.7 Simulasi dan Validasi Sistem Prediksi Cuaca

Simulasi dan validasi sistem prediksi cuaca akan dilakukan setelah algoritma selesai dirancang. Simulasi prediksi cuaca pada wilayah Bandar Udara Internasional Juanda dilakukan dengan data cuaca rata-rata dalam satu hari menggunakan kedua metode, yaitu metode jaringan saraf tiruan dan metode gabungan antara jaringan saraf tiruan dengan *particle swarm optimization*. Simulasi yang dilakukan adalah dengan cara melakukan pelatihan data cuaca untuk memperoleh bobot yang optimum dan tingkat akurasi yang tinggi dari metode yang telah digunakan. Hal tersebut dapat dilihat dari nilai RMSE yang dihasilkan kedua metode tersebut dengan variasi jumlah data pelatihan dan jumlah *hidden node* yang digunakan.

3.8 Analisa Hasil

Analisa hasil pengujian dilakukan setelah melakukan simulasi dan validasi pada sistem prediksi cuaca yang telah dirancang. Sistem prediksi cuaca dapat dikatakan telah berhasil dilakukan atau dapat bekerja dengan baik apabila nilai RMSE dari prediksi yang dihasilkan bernilai kurang dari satu. Prediksi cuaca

yang dapat dikatakan memiliki akurasi yang lebih tinggi dan hasil yang lebih baik apabila nilai RMSE yang dihasilkan semakin mendekati nilai nol.

3.9 Pembahasan Hasil

Hasil yang akan dibahas merupakan perbandingan nilai RMSE antara masing-masing variasi jumlah *hidden node* antara kedua metode tersebut. Metode tersebut dapat dikatakan akurat dan menghasilkan bobot yang optimum dapat dilihat dengan menggunakan nilai RMSE. Tingkat akurasi yang tinggi dan bobot yang optimum dapat dilihat dari nilai RMSE yang lebih kecil diantara kedua metode tersebut.

3.10 Kesimpulan dan Saran

Bagian kesimpulan menunjukkan apakah masalah yang dirumuskan telah dapat terselesaikan atau tidak. Selain itu, kesimpulan juga merupakan jawaban dari rumusan masalah yang telah dibuat. Saran yang dapat digunakan untuk melakukan penelitian selanjutnya yang berhubungan dengan masalah yang sama dengan penelitian tugas akhir terkait kemudian dapat ditulis juga pada subbab kesimpulan dan saran.

Halaman ini sengaja dikosongkan

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pengujian Data Cuaca

Uji dan analisa dilakukan dengan menggunakan *software* IBM SPSS *Sta-tistics* 24, seperti yang sudah dijelaskan pada bab sebelumnya. Berikut merupakan hasil dari analisa data yang digunakan untuk memilih variabel masukan untuk melakukan prediksi kecepatan angin dan visibilitas (jarak pandang).

4.1.1 Analisa Regresi Antar Variabel Cuaca

Analisa regresi digunakan untuk memprediksikan seberapa jauh perubahan nilai variabel dependen (terikat), bila nilai variabel independen (bebas) dimanipulasi atau diubah (Sugiyono, 2017). Hal ini nantinya akan digunakan untuk menentukan variabel cuaca apa saja yang dapat dijadikan sebagai data masukan dalam melakukan prediksi variabel cuaca yang diinginkan, khususnya variabel kecepatan angin dan visibilitas.

- Kecepatan Angin

Variabel dependen pertama yang digunakan merupakan keluaran hasil prediksi pertama yaitu kecepatan angin (WDSP). Variabel independen yang digunakan adalah sebuah variabel cuaca yang telah diperoleh datanya, yaitu TEMP, DEWP, SLP, STP, MXSPD, MAX T., MIN T., dan VISIB. Analisa regresi yang dilakukan adalah analisa regresi linear berganda, yang mana berguna untuk menganalisa hubungan antara banyak variabel independen dengan satu variabel dependen (Priyatno, 2009). Hasil analisa regresi dapat dilihat pada tabel 4.1 dan 4.2.

Tabel 4.1. Analisa Regresi Antara Seluruh Variabel Cuaca dengan Kecepatan Angin

<i>Dependent Variable</i>	R	R²	<i>Std. Error of the Estimate</i>
WDSP	0.773	0.597	1.2206

Nilai R pada tabel 4.1. menunjukkan koefisien korelasi berganda, yaitu antara variabel independen dengan dependen. Jika nilai R mendekati 1, maka hubungannya semakin tinggi. Dengan nilai R sebesar 0,773, dapat disimpulkan bahwa seluruh variabel independen memiliki hubungan yang erat dengan kecepatan angin.

Nilai R^2 menunjukkan koefisien determinasi dengan diubah ke bentuk persen, sehingga menunjukkan persentasi pengaruh variabel independen terhadap variabel dependen. Dengan nilai R^2 sebesar 0,597, artinya persentasi pengaruh seluruh variabel independen terhadap kecepatan angin adalah sebesar 59,7%, sedangkan sisanya sebesar 40,3% dipengaruhi oleh variabel lain yang tidak dimasukkan.

Standard Error of the Estimate menunjukkan ukuran kesalahan prediksi. Dalam kasus ini nilainya sebesar 1,2206 yang berarti kesalahan yang dapat terjadi dalam melakukan prediksi kecepatan angin adalah sebesar 1,2206 Knot.

Tabel 4.2. Analisa Regresi Antara Masing-Masing Variabel Cuaca dengan Kecepatan Angin

<i>Independent Variable</i>	<i>Unstandardized Coefficients</i>		<i>Standardized Coefficients</i>	<i>t</i>	<i>Sig.</i>
	<i>B</i>	<i>Std. Error</i>	<i>Beta</i>		
(Constant)	12.175	21.581	-	0.564	0.573
TEMP(X1)	0.132	0.027	0.145	4.938	0.000
DEWP(X2)	-0.163	0.018	-0.235	-9.266	0.000
SLP(X3)	-1.023	0.189	-0.858	-5.424	0.000
STP(X4)	1.018	0.184	0.871	5.547	0.000
MXSPD(X5)	0.345	0.008	0.655	42.211	0.000
MAX T.(X6)	-0.115	0.019	-0.141	-6.173	0.000
MIN T.(X7)	0.065	0.022	0.082	2.910	0.004
VISIB(X8)	0.410	0.051	0.147	8.005	0.000
*Dependent Variable: WDSP (Y')					

Koefisien B pada tabel 4.2. yang terdiri dari nilai konstan (*constant*) sebesar 12,175, menunjukkan nilai variabel dependen jika seluruh variabel independen bernilai nol, dan juga terdapat koefisien regresi yang terletak dibawah nilai konstan pada tabel. Nilai-nilai ini kemudian dimasukkan ke dalam persamaan linear berganda, sebagai berikut.

$$Y' = 12.125 + 0.132X_1 - 0.163X_2 - 1.023X_3 + 1.018X_4 + 0.345X_5 - 0.115X_6 + 0.065X_7 + 0.410X_8 \quad (4.1)$$

Nilai beta merupakan koefisien yang menunjukkan besarnya pengaruh masing-masing variabel independen terhadap variabel kecepatan angin. Semakin nilai beta mendekati nol, maka pengaruhnya terhadap kecepatan angin semakin kecil. Dilihat dari nilai yang tertera pada tabel, variabel yang memiliki pengaruh yang besar terhadap kecepatan angin dengan nilai koefisien beta diatas 0,2 adalah STP, SLP, MXSPD, dan DEWP.

- Visibilitas (Jarak Pandang)

Variabel dependen kedua yang digunakan merupakan keluaran hasil prediksi kedua yaitu visibilitas (VISIB). Variabel independen yang digunakan adalah sebuah variabel cuaca yang telah diperoleh datanya, yaitu TEMP, DEWP, SLP, STP, MXSPD, MAX T., MIN T., dan WDSP. Analisa regresi yang dilakukan adalah analisa regresi linear berganda, yang mana berguna untuk menganalisa hubungan antara banyak variabel independen dengan satu variabel dependen (Priyatno, 2009). Hasil analisa regresi dapat dilihat pada tabel 4.3 dan 4.4.

Tabel 4.3. Analisa Regresi Antara Seluruh Variabel Cuaca dengan Visibilitas

<i>Dependent Variable</i>	R	R²	<i>Std. Error of the Estimate</i>
VISIB	0.600	0.361	0.358

Nilai R pada tabel 4.3. menunjukkan koefisien korelasi berganda, yaitu antara variabel independen dengan variabel dependen. Jika nilai R mendekati 1, maka hubungannya semakin tinggi. Dengan nilai R sebesar 0.600, dapat disimpulkan bahwa seluruh variabel independen memiliki hubungan yang cukup erat dengan visibilitas.

Nilai R^2 menunjukkan koefisien determinasi dengan diubah ke bentuk persen, sehingga menunjukkan persentasi pengaruh variabel independen terhadap variabel dependen. Dengan nilai R^2 sebesar 0.361, artinya persentasi pengaruh seluruh variabel independen terhadap visibilitas adalah sebesar 36,1%, sedangkan sisanya sebesar 63,9% dipengaruhi oleh variabel lain yang tidak dimasukkan.

Standard Error of the Estimate menunjukkan ukuran kesalahan prediksi. Dalam kasus ini nilainya sebesar 0,358 yang berarti kesalahan yang dapat terjadi dalam melakukan prediksi visibilitas adalah sebesar 0,358 mil.

Tabel 4.4. Analisa Regresi Antara Masing-Masing Variabel Cuaca dengan Visibilitas

<i>Independent Variable</i>	<i>Unstandardized Coefficients</i>		<i>Standardized Coefficients</i>	<i>t</i>	<i>Sig.</i>
	<i>B</i>	<i>Std. Error</i>	<i>Beta</i>		
(Constant)	2.997	9.725	-	0.308	0.758
TEMP(X1)	0.205	0.011	0.633	18.526	0.000
DEWP(X2)	-0.061	0.008	-0.244	-7.596	0.000
SLP(X3)	1.384	0.079	3.244	17.461	0.000
STP(X4)	-1.386	0.077	-3.315	-18.056	0.000
MXSPD(X5)	-0.018	0.005	-0.096	-3.493	0.000
MAX T.(X6)	-0.092	0.008	-0.316	-11.268	0.000
MIN T.(X7)	-0.017	0.010	-0.058	-1.628	0.104
WDSP(X8)	0.083	0.010	0.233	8.005	0.000

*Dependent Variable: VISIB (Y')

Koefisien B pada tabel 4.4. yang terdiri dari nilai konstan (*constant*) sebesar 2,997, menunjukkan nilai variabel dependen jika seluruh variabel independen bernilai nol, dan juga terdapat koefisien regresi yang terletak dibawah nilai konstan pada tabel. Nilai-nilai ini kemudian dimasukkan ke dalam persamaan linear berganda, sebagai berikut.

$$Y' = 2.997 + 0.205X_1 - 0.061X_2 + 1.384X_3 - 1.386X_4 - 0.018X_5 - 0.092X_6 - 0.017X_7 + 0.083X_8 \quad (4.2)$$

Nilai beta merupakan koefisien yang menunjukkan besarnya pengaruh masing-masing variabel independen terhadap variabel visibilitas. Semakin nilai beta mendekati nol, maka pengaruhnya terhadap visibilitas semakin kecil. Dilihat dari nilai yang tertera pada tabel, variabel yang memiliki pengaruh yang besar terhadap kecepatan angin dengan nilai koefisien beta diatas 0,2 adalah STP, SLP, TEMP, MAX T., DEWP, dan WDSP.

4.1.2 Analisa Korelasi Antar Variabel Cuaca

Analisa korelasi dilakukan untuk melihat hubungan antara satu variabel cuaca dengan variabel cuaca lainnya yang dapat dilihat dari nilai koefisien korelasi. Dalam hal ini, analisa korelasi digunakan untuk melihat hubungan antar variabel independen sehingga dapat diketahui variabel independen mana saja yang dapat digunakan secara bersama-sama sebagai data masukan untuk melakukan prediksi. Interval koefisien korelasi antara 0 hingga 0,199 tingkat korelasinya sangat rendah, 0,2 hingga 0,399 tingkat korelasinya rendah, 0,4 hingga 0,599 tingkat korelasinya sedang, 0,6 hingga 0,799 tingkat korelasinya kuat, dan 0,8 hingga 1 tingkat korelasinya sangat kuat (Sugiyono, 2017).

- Kecepatan Angin

Terdapat empat variabel yang memiliki pengaruh besar terhadap perubahan kecepatan angin dilihat dari hasil analisa regresi, yaitu STP dengan koefisien beta 0,871, SLP dengan koefisien beta sebesar -0,858, MXSPD dengan koefisien beta

0,655, dan DEWP dengan koefisien beta -0,235. Kemudian dapat dilihat koefisien korelasi antar variabel tersebut pada tabel 4.5.

Tabel 4.5. Koefisien Korelasi dengan Variabel Dependen Kecepatan Angin

	DEWP	SLP	STP	MXSPD
DEWP	1	-0.463	-0.442	-0.236
SLP	-0.463	1	0.994	0.095
STP	-0.442	0.994	1	0.094
MXSPD	-0.236	0.095	0.094	1

Tabel 4.5 menjelaskan bahwa nilai koefisien korelasi yang dihasilkan antara DEWP, SLP, dan STP masing-masing berada pada interval tingkat korelasi sedang dan sangat kuat. Sedangkan, nilai antara MXSPD dengan DEWP, SLP, dan STP berada pada tingkat korelasi rendah dan sangat rendah. Tetapi, dapat dilihat juga koefisien beta antara MXSPD dengan kecepatan angin (WDSP) bernilai cukup besar dibanding variabel lainnya yaitu sebesar 0,655. Sehingga, dapat disimpulkan variabel DEWP, SLP, STP, dan MXSPD tetap dapat digunakan sebagai variabel masukan untuk melakukan prediksi kecepatan angin.

- Visibilitas (Jarak Pandang)

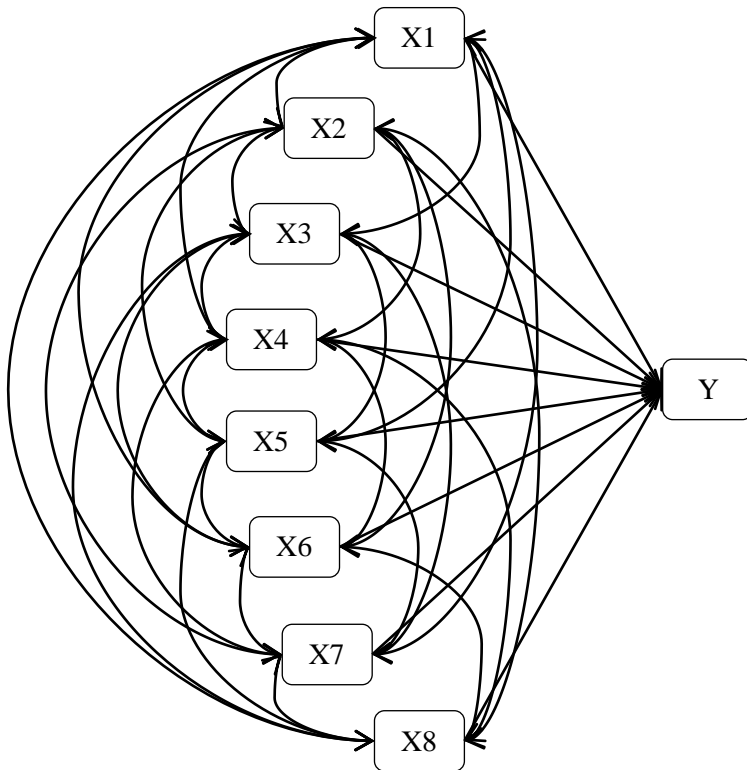
Terdapat enam variabel yang memiliki pengaruh besar terhadap perubahan kecepatan angin dilihat dari hasil analisa regresi, yaitu STP dengan koefisien beta -3,315, SLP dengan koefisien beta 3,244, TEMP dengan koefisien beta 0,633, MAX T. dengan koefisien beta -0,316, DEWP dengan koefisien beta bernilai -0,244, dan WDSP dengan koefisien beta bernilai 0,233. Kemudian dapat dilihat koefisien korelasi antar variabel tersebut pada tabel 4.6.

Tabel 4.6. Koefisien Korelasi dengan Variabel Dependen Visibilitas

	TEMP	DEWP	SLP	STP	MAX	WDSP
TEMP	1	0.301	-0.129	-0.104	0.715	0.087
DEWP	0.301	1	-0.463	-0.442	0.109	-0.321
SLP	-0.129	-0.463	1	0.994	-0.118	0.155
STP	-0.104	-0.442	0.994	1	-0.099	0.157
MAX T.	0.715	0.109	-0.118	-0.099	1	0.002
WDSP	0.087	-0.321	0.155	0.157	0.002	1

Tabel 4.6 menjelaskan bahwa nilai koefisien korelasi yang dihasilkan antara DEWP, SLP, dan STP masing-masing berada pada interval tingkat korelasi sedang dan sangat kuat. Sedangkan, nilai korelasi antara TEMP, MAX T., dan WDSP rata-rata terdapat pada interval rendah dan sangat rendah, hanya korelasi antara TEMP dan MAX T. saja yang berada pada tingkat korelasi kuat. Tetapi, dapat dilihat bahwa nilai koefisien beta variabel TEMP cukup besar jika dibandingkan dengan variabel MAX T. dan WDSP. Dengan kata lain, variabel MAX T. dan WDSP memiliki tingkat korelasi yang rendah dengan variabel lainnya, dan memiliki pengaruh yang cukup kecil terhadap variabel visibilitas. Sehingga dapat disimpulkan bahwa hanya variabel TEMP, DEWP, SLP, dan STP yang dapat digunakan sebagai variabel masukan untuk melakukan prediksi visibilitas.

4.1.3 Analisa Jalur (*Path Analysis*) Antar Variabel Cuaca



Gambar 4.1. Hipotesis awal diagram jalur antar variabel cuaca

Analisa jalur digunakan untuk untuk menunjukkan hubungan (korelasi) dan besar pengaruh (regresi) antar variabel yang berbentuk sebab akibat atau kausal yang dalam hal ini adalah variabel independen dan variabel dependen. Analisa jalur dilakukan dengan melihat hasil analisa regresi dan korelasi yang telah dilakukan sebelumnya dan kemudian membuat hipotesis awal diagram jalur. Kemudian berdasarkan besar koefisien beta pada analisa regresi dan besar koefisien korelasi pada analisa korelasi, maka dapat dipilih variabel cuaca apa saja yang dapat digunakan

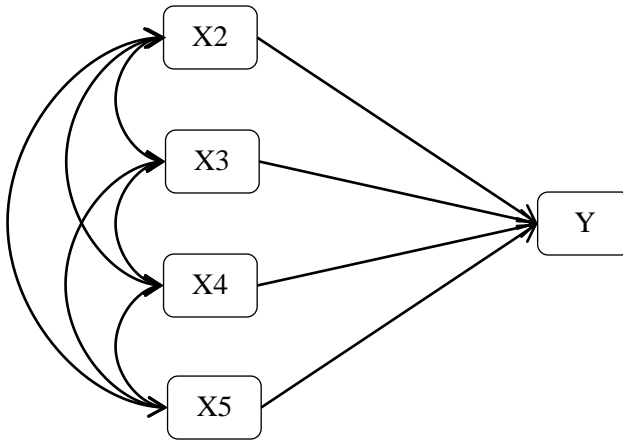
sebagai masukan untuk melakukan prediksi kecepatan angin dan visibilitas.

- Kecepatan Angin

Hipotesis awal diagram jalur untuk melakukan prediksi kecepatan angin (Y), yang digunakan sebagai masukan atau dapat disebut juga variabel kausalnya adalah TEMP (X1), DEWP (X2), SLP (X3), STP (X4), MXSPD (X5), MAX T. (X6), MIN T. (X7), dan VISIB (X8). Berikut ini merupakan kesimpulan dari hasil analisa jalur untuk variabel dependen kecepatan angin:

- a. Pengaruh variabel DEWP terhadap WDSP adalah sebesar -0,235 atau -23,5%.
- b. Pengaruh variabel SLP terhadap WDSP adalah sebesar -0,858 atau 85,8%.
- c. Pengaruh variabel STP terhadap WDSP adalah sebesar 0,871 atau 87,1%.
- d. Pengaruh variabel MXSPD terhadap WDSP adalah sebesar 0,655 atau 65,5%.
- e. Pengaruh variabel independen secara gabungan terhadap variabel dependen (WDSP) adalah sebesar 0,597 atau 59,7%.
- f. Pengaruh variabel lain di luar analisa jalur adalah sebesar 40,3%
- g. Korelasi antara variabel DEWP dengan SLP adalah sebesar -0,463
- h. Korelasi antara variabel DEWP dengan STP adalah sebesar -0,442
- i. Korelasi antara variabel DEWP dengan MXSPD adalah sebesar -0,236
- j. Korelasi antara variabel SLP dengan STP adalah sebesar 0,994
- k. Korelasi antara variabel SLP dengan MXSPD adalah sebesar 0,095
- l. Korelasi antara variabel STP dengan MXSPD adalah sebesar 0,094

Setelah dilakukan analisa jalur dengan menggunakan analisa regresi dan korelasi, maka diperoleh hasil diagram analisa jalur seperti pada gambar 4.2.



Gambar 4.2. Diagram jalur untuk prediksi kecepatan angin

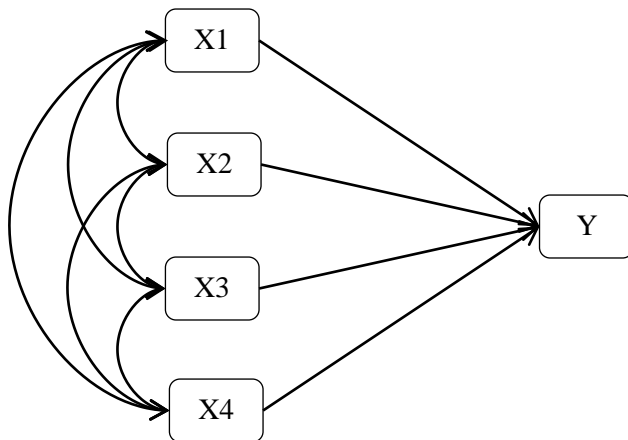
- Visibilitas (Jarak Pandang)

Hipotesis awal diagram jalur untuk melakukan prediksi visibilitas (Y), yang digunakan sebagai masukan atau dapat disebut juga variabel kausalnya adalah TEMP (X1), DEWP (X2), SLP (X3), STP (X4), MXSPD (X5), MAX T. (X6), MIN T. (X7), dan WDSP (X8). Berikut ini merupakan kesimpulan dari hasil analisa jalur untuk variabel dependen kecepatan angin:

- Pengaruh variabel TEMP terhadap VISIB adalah sebesar 0,633 atau 63,3%.
- Pengaruh variabel DEWP terhadap VISIB adalah sebesar - 0,244 atau 24,4%.
- Pengaruh variabel SLP terhadap VISIB adalah sebesar 3,244 atau 324,4%.
- Pengaruh variabel STP terhadap VISIB adalah sebesar 3,315 atau 331,5%.
- Pengaruh variabel independen secara gabungan terhadap variabel dependen (VISIB) adalah sebesar 0,361 atau 36,1%.

- f. Pengaruh variabel lain di luar analisa jalur adalah sebesar 63,9%
- g. Korelasi antara variabel TEMP dengan DEWP adalah sebesar 0,301
- h. Korelasi antara variabel TEMP dengan SLP adalah sebesar -0,129
- i. Korelasi antara variabel TEMP dengan STP adalah sebesar -0,104
- j. Korelasi antara variabel DEWP dengan SLP adalah sebesar -0,463
- k. Korelasi antara variabel DEWP dengan STP adalah sebesar -0,422
- l. Korelasi antara variabel SLP dengan STP adalah sebesar 0,994

Setelah dilakukan analisa jalur dengan menggunakan analisa regresi dan korelasi, maka diperoleh hasil diagram analisa jalur seperti pada gambar 4.3.



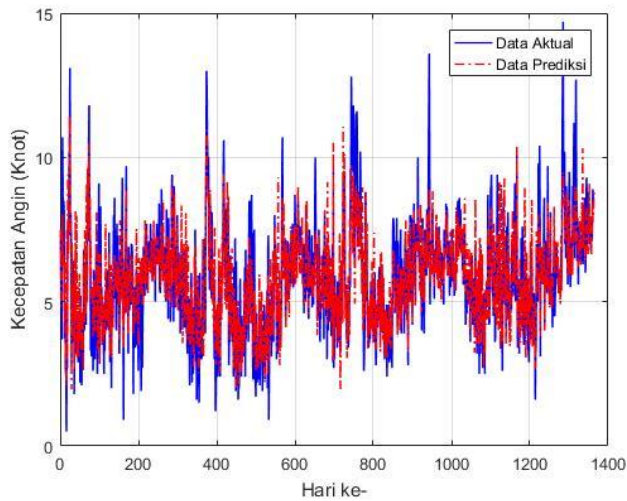
Gambar 4.3. Diagram jalur untuk prediksi visibilitas

4.2 Prediksi Cuaca dengan menggunakan Jaringan Saraf Tiruan (JST)

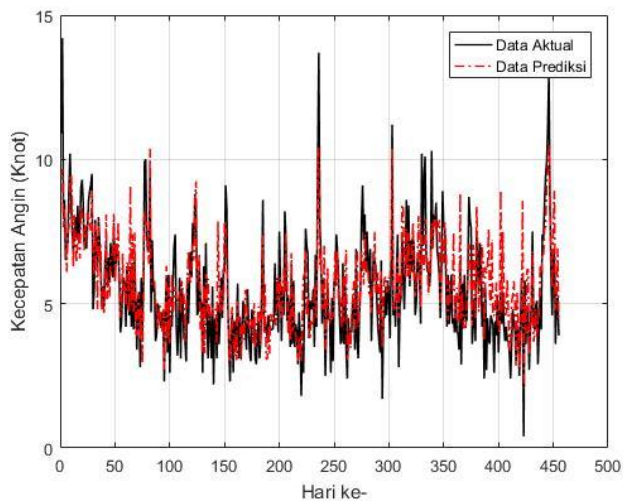
Jumlah *hidden node* pada arsitektur jaringan saraf tiruan dapat mempengaruhi performa dan keakuratan dari hasil keluaran jaringan (Nugraha & SN, 2014). Pada penelitian ini digunakan arsitektur jaringan saraf tiruan dengan menggunakan satu *hidden layer*, satu *input layer*, dan satu *output layer*. Variabel yang digunakan pada *input layer* dipilih berdasarkan hasil dari pengujian data cuaca sebelumnya. Di dalam *hidden layer* terdapat beberapa *hidden node*, yang mana jumlah *hidden node* nantinya akan mempengaruhi hasil pelatihan dan validasi dari prediksi cuaca yang akan dilakukan. Data yang digunakan adalah data cuaca dari tahun 2012 hingga 2016 sebanyak 1821 set data. Data tersebut dibagi menjadi dua bagian, yaitu sebanyak 1365 set data untuk data pelatihan dan 456 set data untuk validasi data. Untuk mengetahui berapa jumlah *hidden node* yang akan menghasilkan nilai terbaik, dilakukan percobaan dengan menggunakan *hidden node* dengan jumlah 1 hingga 10 dengan menggunakan algoritma jaringan saraf tiruan. Dari percobaan tersebut kemudian dianalisa hasil yang dapat dilihat dari nilai RMSE pelatihan data, RMSE validasi data prediksi, dan performansi pada setiap iterasinya. Dari hasil tersebut kemudian dapat dipilih jumlah *hidden node* yang tepat untuk digunakan dalam melakukan prediksi kecepatan angin dan visibilitas.

4.2.1 Pelatihan dan Validasi untuk Memprediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan

Variabel yang dapat digunakan sebagai variabel masukan pada *input layer* untuk melakukan prediksi kecepatan angin berdasarkan hasil pengujian data adalah DEWP, SLP, STP, dan MXSPD. Kemudian dilakukan pelatihan dan validasi data untuk prediksi kecepatan angin (MXSPD) dengan hasil terbaik yang diperoleh adalah *hidden node* berjumlah 3 *node* yang dapat dilihat pada grafik dibawah ini.



Gambar 4.4. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 3 *hidden node*



Gambar 4.5. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 3 *hidden node*

Nilai RMSE untuk pelatihan dan validasi data dari prediksi kecepatan angin menggunakan jaringan saraf tiruan dengan variasi jumlah *hidden node* dapat dilihat pada tabel 4.7.

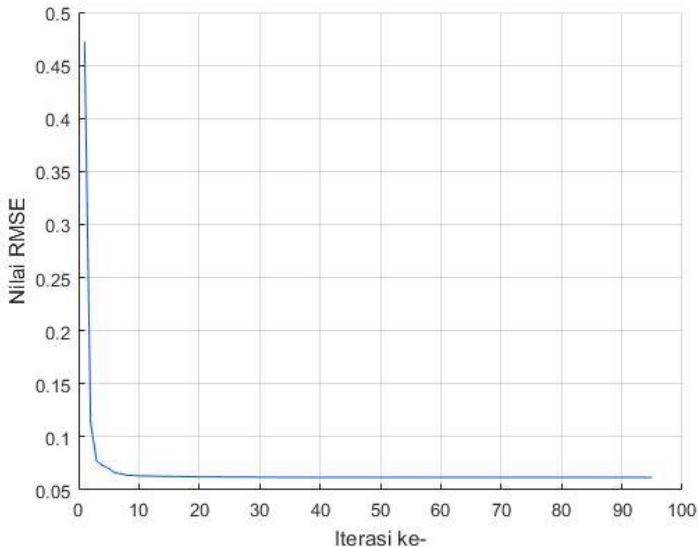
Tabel 4.7. Nilai RMSE Prediksi Kecepatan Angin dengan Menggunakan Jaringan Saraf Tiruan

Jumlah <i>Hidden Node</i>	Metode JST	
	Nilai RMSE	
	Pelatihan	Validasi
1	0.063848	0.071623
2	0.063198	0.070844
3	0.0618	0.066912
4	0.06125	0.067993
5	0.06014	0.067435
6	0.059495	0.067475
7	0.05922	0.067245
8	0.058956	0.069406
9	0.058094	0.071229
10	0.057183	0.079567

Tabel 4.7. menjelaskan hasil prediksi berupa nilai RMSE validasi, yang memiliki nilai terendah yaitu sebesar 0,066912 terdapat pada jumlah *hidden node* yang digunakan sebanyak 3 *node*. Jumlah *hidden node* tersebut juga dapat dilihat grafik prediksi yang dihasilkan sudah hampir mendekati grafik cuaca aktual. Kesimpulan yang dapat diambil yaitu dengan menggunakan 3 *hidden node* dapat dilakukan prediksi kecepatan angin menggunakan jaringan saraf tiruan.

4.2.2 Performansi Pelatihan Data Kecepatan Angin Menggunakan Jaringan Saraf Tiruan

Algoritma pelatihan yang digunakan pada prediksi kecepatan angin menggunakan metode jaringan saraf tiruan adalah algoritma Levenberg-Marquardt. Sesuai fungsinya, algoritma pelatihan akan melakukan pembaharuan bobot pada tiap iterasinya hingga menghasilkan nilai RMSE paling kecil yang dapat diperoleh, dengan jumlah iterasi yang digunakan sebanyak 150. Grafik performansi pada gambar dibawah ini menunjukkan perubahan nilai RMSE pelatihan data kecepatan angin pada setiap iterasi.



Gambar 4.6. Grafik performansi pelatihan JST untuk data kecepatan angin

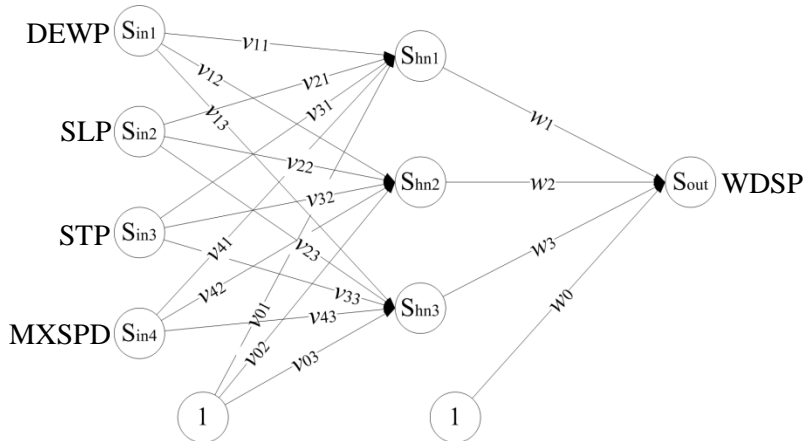
Grafik 4.6. menjelaskan perubahan nilai RMSE yang semakin menurun dari iterasi pertama hingga iterasi ke-34. Setelah iterasi ke-34, nilai RMSE menunjukkan tidak ada perubahan atau stabil. Tidak adanya perubahan nilai RMSE juga berarti tidak adanya pembaharuan bobot, karena bobot yang dihasilkan pada

iterasi tersebut sudah konvergen untuk mencapai nilai RMSE yang terkecil.

4.2.3 Jumlah *Hidden Node* untuk Prediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan

Variabel masukan yang tepat untuk melakukan prediksi kecepatan angin telah diperoleh dari hasil pengujian data. Hasil percobaan sebelumnya juga telah memperoleh jumlah *hidden node* yang menghasilkan nilai RMSE terkecil, yang kemudian dapat dikatakan memiliki akurasi yang paling tinggi. Arsitektur jaringan terbaik juga telah diperoleh dari hasil pelatihan dan validasi data. Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah. Rendahnya nilai RMSE menunjukkan bahwa arsitektur tersebut dapat digunakan untuk menghasilkan akurasi prediksi cuaca yang paling tinggi dibandingkan jumlah *hidden node* lainnya.

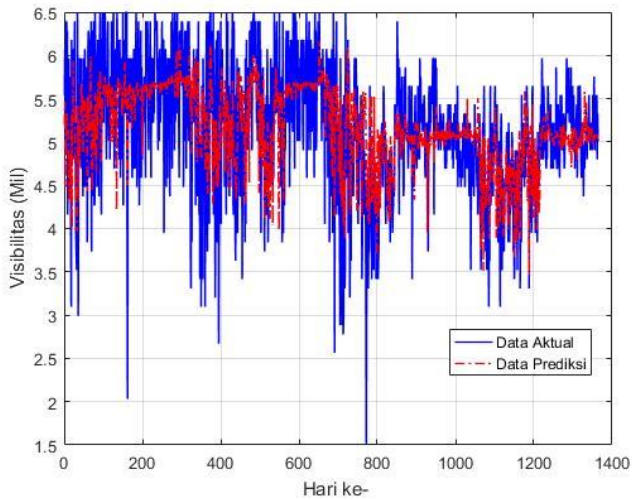
Arsitektur jaringan saraf tiruan diperoleh dari hasil pengujian data yang digunakan untuk menentukan variabel apa saja yang digunakan sebagai masukan, dan pengujian jumlah *hidden node* yang digunakan untuk menentukan jumlah *hidden node* yang menghasilkan RMSE terkecil. Sehingga diperoleh masukan jaringan saraf tiruan untuk prediksi kecepatan angin berjumlah 4 variabel yaitu DEWP, SLP, STP, dan MXSPD. Selain itu, untuk jumlah variabel diambil dari hasil pengujian jumlah *hidden node* yang memiliki nilai RMSE terkecil yaitu pada *hidden node* yang berjumlah 3 *node*. Arsitektur jaringan saraf tiruan untuk prediksi kecepatan angin dapat dilihat pada gambar 4.7.



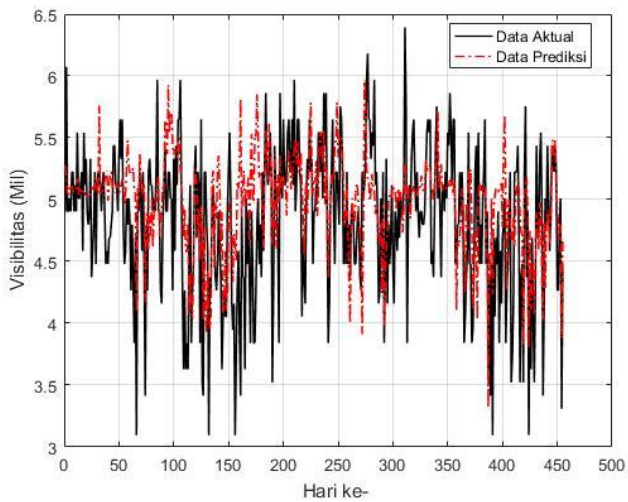
Gambar 4.7. Arsitektur JST untuk prediksi kecepatan angin

4.2.4 Pelatihan dan Validasi untuk Memprediksi Visibilitas (Jarak Pandang) Menggunakan Jaringan Saraf Tiruan

Terdapat empat variabel yang dapat digunakan sebagai variabel masukan pada *input layer* untuk melakukan prediksi visibilitas berdasarkan hasil pengujian data yaitu TEMP, DEWP, SLP dan STP. Pelatihan dan validasi data kemudian dilakukan untuk prediksi visibilitas (VISIB) dengan jumlah *hidden node* yang digunakan tetap bervariasi dari 1 hingga 10 *hidden node*. Hasil grafik pelatihan dan validasi data untuk prediksi visibilitas (VISIB) dengan hasil terbaik yang diperoleh adalah *hidden node* berjumlah 4 *node* yang dapat dilihat pada grafik dibawah ini.



Gambar 4.8. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 4 *hidden node*



Gambar 4.9. Hasil validasi JST untuk data visibilitas dengan menggunakan 4 *hidden node*

Hasil prediksi visibilitas menggunakan jaringan saraf tiruan dengan variasi jumlah *hidden node* dapat dilihat pada tabel 4.8.

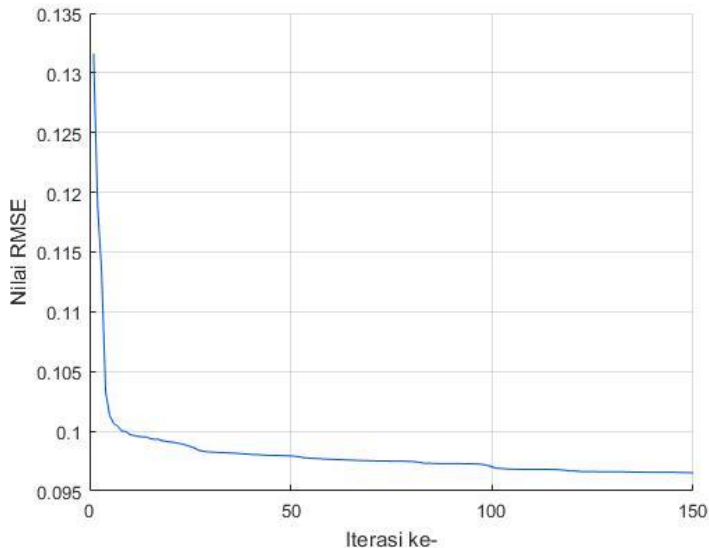
Tabel 4.8. Nilai RMSE Prediksi Visibilitas dengan Menggunakan Jaringan Saraf Tiruan

Jumlah <i>Hidden</i> <i>Node</i>	Metode JST	
	Nilai RMSE	
	Pelatihan	Validasi
1	0.101669	0.088312
2	0.100616	0.090795
3	0.09694	0.085632
4	0.096514	0.085088
5	0.094424	0.085294
6	0.093342	0.088315
7	0.095134	0.089161
8	0.093264	0.089216
9	0.093032	0.08606
10	0.092377	0.085646

Tabel 4.8 menjelaskan hasil prediksi berupa nilai RMSE validasi. Jumlah *hidden node* yang memiliki nilai terendah yaitu sebesar 0,085088 terdapat pada arsitektur dengan menggunakan 4 *node*. Grafik prediksi yang dihasilkan oleh arsitektur yang menggunakan 4 *hidden node* sudah hampir mendekati grafik cuaca aktual. Kesimpulan yang dapat diambil yaitu dapat prediksi visibilitas menggunakan jaringan saraf tiruan dapat dilakukan dengan menggunakan 4 *hidden node*

4.2.5 Performansi Pelatihan Data Visibilitas Menggunakan Jaringan Saraf Tiruan

Prediksi visibilitas menggunakan jaringan saraf tiruan juga menggunakan algoritma pelatihan Levenberg-Marquardt dengan jumlah iterasi yang digunakan sebanyak 150. Grafik performansi pada gambar dibawah ini menunjukkan perubahan nilai RMSE pelatihan data visibilitas pada setiap iterasi.



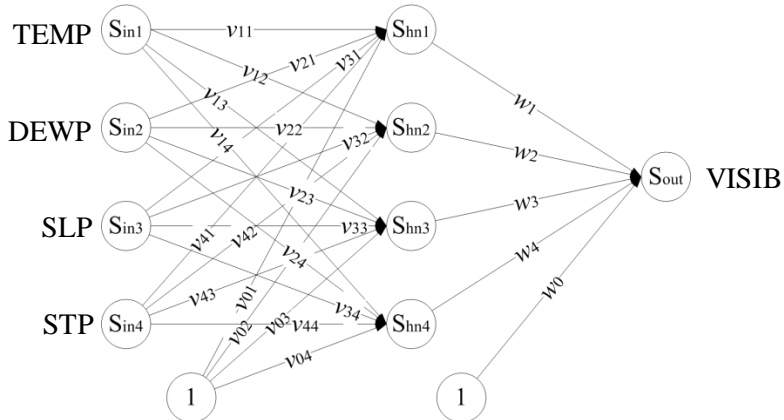
Gambar 4.10. Grafik performansi pelatihan JST untuk data visibilitas

Grafik 4.10. menunjukkan perubahan nilai RMSE yang menurun secara signifikan hingga iterasi ke-50. Setelah iterasi ke-50, nilai RMSE menunjukkan adanya penurunan yang tidak terlalu banyak atau dapat dikatakan mulai stabil. Stabilitasnya nilai RMSE juga berarti tidak terlalu banyak pembaharuan bobot, karena bobot yang dihasilkan pada iterasi tersebut sudah konvergen untuk mencapai nilai RMSE yang terkecil.

4.2.6 Jumlah *Hidden Node* untuk Prediksi Visibilitas (Jarak Pandang) Menggunakan Jaringan Saraf Tiruan

Variabel masukan yang tepat untuk melakukan prediksi visibilitas telah diperoleh dari hasil pengujian data. Lalu, dari hasil percobaan sebelumnya juga telah diperoleh jumlah *hidden node* yang menghasilkan nilai RMSE terkecil yang kemudian dapat dikatakan memiliki akurasi yang paling tinggi. Langkah berikutnya yang dilakukan juga tetap sama dengan percobaan pada prediksi kecepatan angin menggunakan jaringan saraf tiruan. Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah.

Variabel masukan jaringan saraf tiruan untuk prediksi visibilitas berjumlah 4 variabel yang diperoleh yaitu TEMP, DEWP, SLP, dan STP. Lapisan tersembunyi memiliki *hidden node* dengan jumlah yang digunakan diambil dari hasil pengujian jumlah *hidden node* yang memiliki nilai RMSE terkecil yaitu pada *hidden node* yang berjumlah 4 *node*.



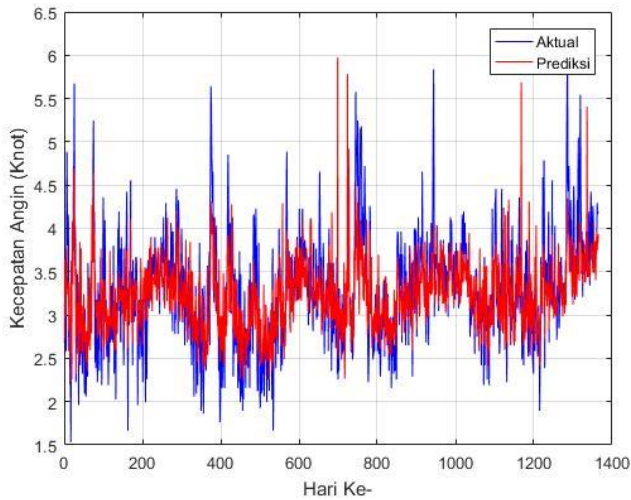
Gambar 4.11. Arsitektur JST untuk prediksi visibilitas

4.3 Prediksi Cuaca dengan menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization* (JST-PSO)

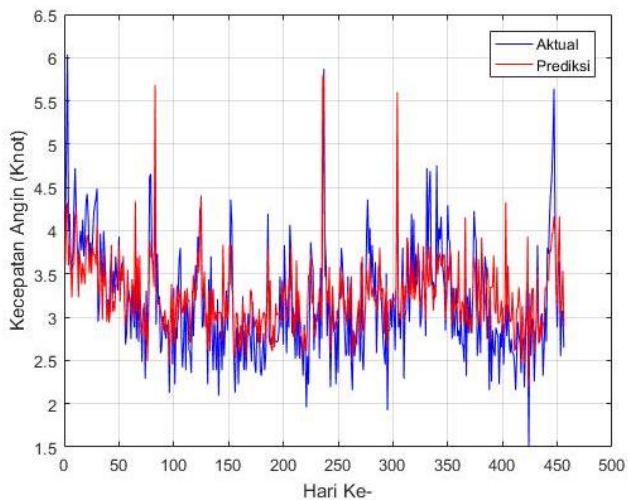
Setelah melakukan prediksi cuaca menggunakan metode jaringan saraf tiruan, berikutnya dilakukan percobaan prediksi cuaca dengan menggunakan metode gabungan jaringan saraf tiruan dan *particle swarm optimization*. Hal ini dilakukan untuk mengetahui apakah metode gabungan dapat meningkatkan akurasi dan memperoleh bobot optimum dengan melihat nilai RMSE yang dihasilkan. Hal yang menjadi perbedaan antara kedua metode ini adalah pada algoritma pelatihannya. Pada metode *artificial neural network* yang digunakan sebagai algoritma pelatihan adalah algoritma Lavenberg-Marquadt, sedangkan pada metode gabungan yang digunakan sebagai algoritma pelatihan adalah metode optimasinya, yang dalam hal ini adalah algoritma *Particle Swarm Optimization*. Sama dengan metode sebelumnya, percobaan ini dilakukan dengan menggunakan variasi jumlah *hidden node* yang sama dengan metode jaringan saraf tiruan yaitu berjumlah 1 hingga 10 *node*.

4.3.1 Pelatihan dan Validasi untuk Memprediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Variabel masukan yang digunakan pada percobaan ini tetap menggunakan hasil dari pengujian data, yaitu menggunakan variabel DEWP, SLP, STP, dan MXSPD untuk melakukan prediksi kecepatan angin. Kemudian dilakukan pelatihan dan validasi data untuk prediksi kecepatan angin (MXSPD) dengan menggunakan metode gabungan. Hasil terbaik yang diperoleh adalah *hidden node* berjumlah 6 *node* yang dapat dilihat pada grafik dibawah ini.



Gambar 4.12. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 6 *hidden node*



Gambar 4.13. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 6 *hidden node*

Hasil prediksi kecepatan angin dengan menggunakan metode gabungan jaringan saraf tiruan dan *particle swarm optimization* dengan variasi jumlah *hidden node* dapat dilihat pada tabel 4.9.

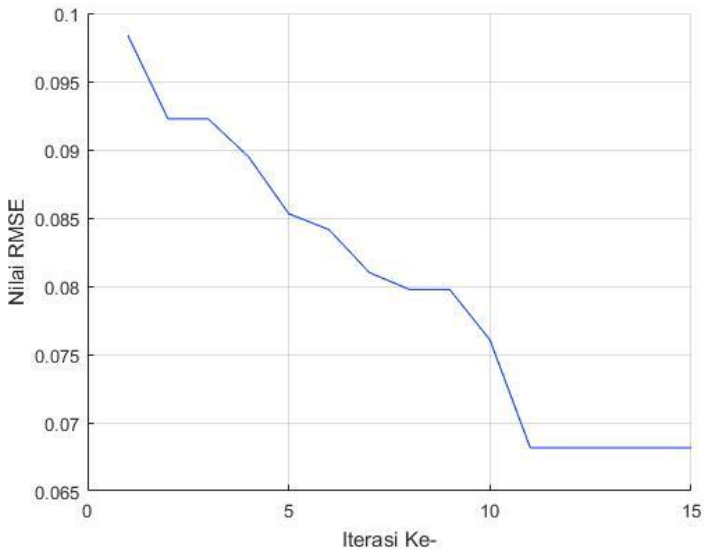
Tabel 4.9. Nilai RMSE Prediksi Kecepatan Angin dengan Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Jumlah <i>Hidden Node</i>	Metode JST-PSO	
	Nilai RMSE	
	Pelatihan	Validasi
1	0.0825	0.0826
2	0.0804	0.0778
3	0.0784	0.0762
4	0.0923	0.0954
5	0.0987	0.1
6	0.0678	0.0721
7	0.0817	0.0797
8	0.0887	0.0855
9	0.0794	0.0778
10	0.0923	0.0858

Tabel 4.9 menjelaskan hasil prediksi yang berupa nilai RMSE. Nilai RMSE terendah yaitu sebesar 0,0721 terdapat pada arsitektur jaringan saraf tiruan dengan jumlah *hidden node* sebanyak 6 *node*. Jika dibandingkan dengan metode tanpa optimisasi, nilai RMSE yang dihasilkan oleh metode gabungan lebih besar tetapi masih bisa mendekati nilai RMSE yang tanpa menggunakan optimisasi. Kesimpulan yang dapat diambil yaitu bahwa metode gabungan jaringan saraf tiruan dan *particle swarm optimizat*on dapat digunakan untuk melakukan prediksi cuaca dan dapat menghasilkan bobot yang optimum.

4.3.2 Performansi Pelatihan Data Kecepatan Angin Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Algoritma pelatihan yang digunakan pada metode gabungan adalah menggunakan *particle swarm optimization*. *Particle swarm optimization* akan melakukan pembaharuan bobot hingga mencapai nilai RMSE minimum, dengan jumlah iterasi sebanyak 15 iterasi. Grafik hasil optimisasi atau dapat disebut juga grafik performansi pelatihan dengan menggunakan *particle swarm optimization* untuk prediksi kecepatan angin dapat dilihat pada gambar dibawah ini.



Gambar 4.14. Grafik performansi pelatihan JST-PSO untuk data kecepatan angin

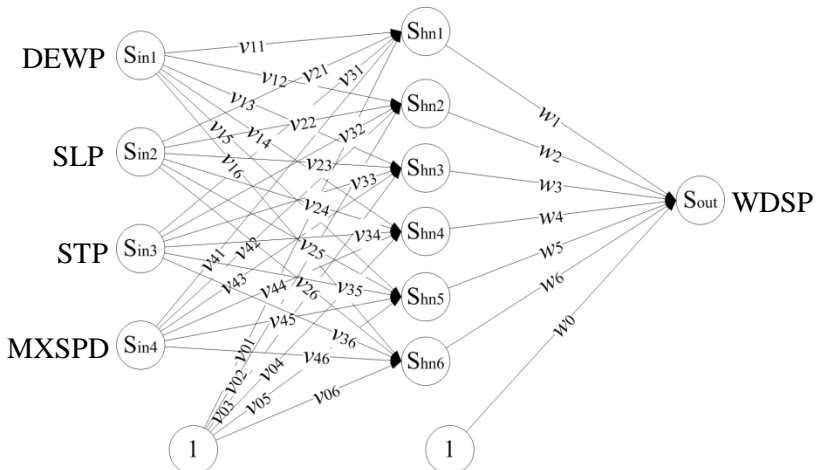
Grafik pada gambar 4.14. menunjukkan bahwa nilai bobot yang terdapat pada jaringan saraf tiruan dapat diperbaharui dan menghasilkan nilai RMSE yang semakin kecil seiring dengan bertambahnya iterasi. Sehingga dapat disimpulkan bahwa *particle*

swarm optimization dapat digunakan untuk melakukan pencarian bobot optimum pada jaringan saraf tiruan.

4.3.3 Jumlah *Hidden Node* untuk Prediksi Kecepatan Angin Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Variabel masukan yang tepat untuk melakukan prediksi kecepatan angin dari hasil pengujian data. Lalu, dari hasil percobaan sebelumnya juga telah diperoleh jumlah *hidden node* yang menghasilkan nilai RMSE terkecil yang kemudian dapat dikatakan memiliki akurasi yang paling tinggi. Dari hasil tersebut kemudian diperoleh arsitektur jaringan terbaik dari hasil pelatihan dan validasi data. Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah. Rendahnya nilai RMSE menunjukkan bahwa arsitektur tersebut dapat digunakan untuk menghasilkan akurasi prediksi cuaca yang paling tinggi dibandingkan jumlah *hidden node* lainnya.

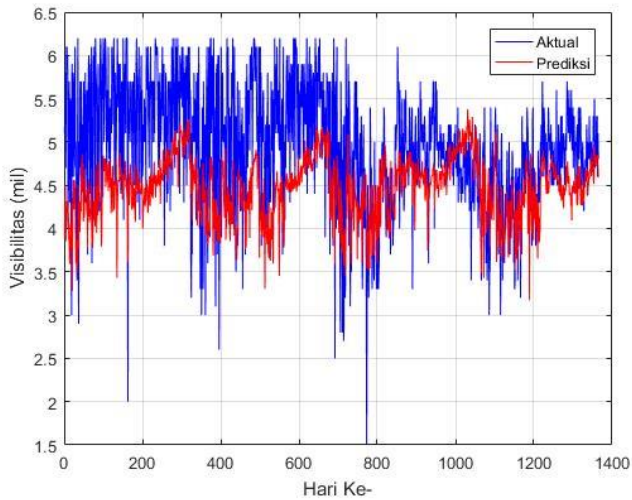
Arsitektur jaringan saraf tiruan diperoleh dari hasil pengujian data yang digunakan untuk menentukan variabel apa saja yang digunakan sebagai masukan, dan pengujian jumlah *hidden node* yang digunakan untuk menentukan jumlah *hidden node* yang menghasilkan RMSE terkecil. Sehingga diperoleh masukan jaringan saraf tiruan untuk prediksi kecepatan angin berjumlah 4 variabel yaitu DEWP, SLP, STP, dan MXSPD. Selain itu, untuk jumlah variabel diambil dari hasil pengujian jumlah *hidden node* yang memiliki nilai RMSE terkecil yaitu pada *hidden node* yang berjumlah 6 *node*. Arsitektur jaringan saraf tiruan untuk prediksi kecepatan angin dengan menggunakan metode gabungan jaringan saraf tiruan dan *particle swarm optimization* dapat dilihat pada gambar dibawah ini.



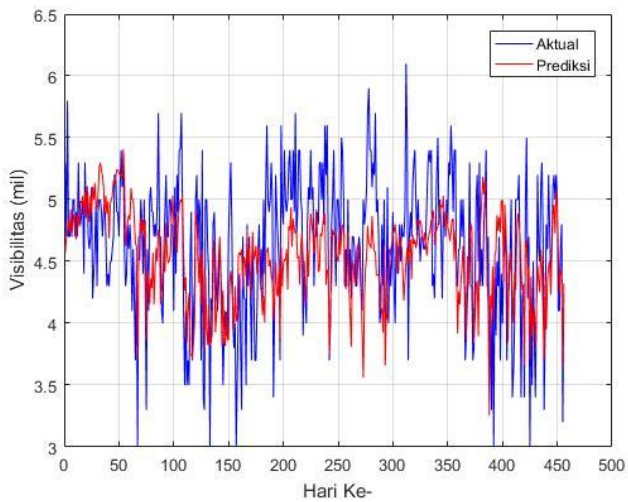
Gambar 4.15. Arsitektur JST-PSO untuk prediksi kecepatan angin

4.3.4 Pelatihan dan Validasi untuk Memprediksi Visibilitas Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Variabel masukan yang digunakan pada percobaan ini tetap menggunakan hasil dari pengujian data, yaitu menggunakan variabel TEMP, DEWP, SLP, dan STP untuk melakukan prediksi visibilitas. Kemudian dilakukan pelatihan dan validasi data untuk prediksi visibilitas (VISIB) dengan menggunakan metode gabungan dan jumlah *hidden node* yang digunakan tetap bervariasi dari 1 hingga 10 *hidden node*. Hasil terbaik yang diperoleh adalah *hidden node* berjumlah 9 *node* yang dapat dilihat pada grafik dibawah ini..



Gambar 4.16. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 9 *hidden node*



Gambar 4.17. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 9 *hidden node*

Hasil prediksi visibilitas dengan menggunakan metode gabungan jaringan saraf tiruan dan *particle swarm optimization* dapat dilihat pada tabel 4.10.

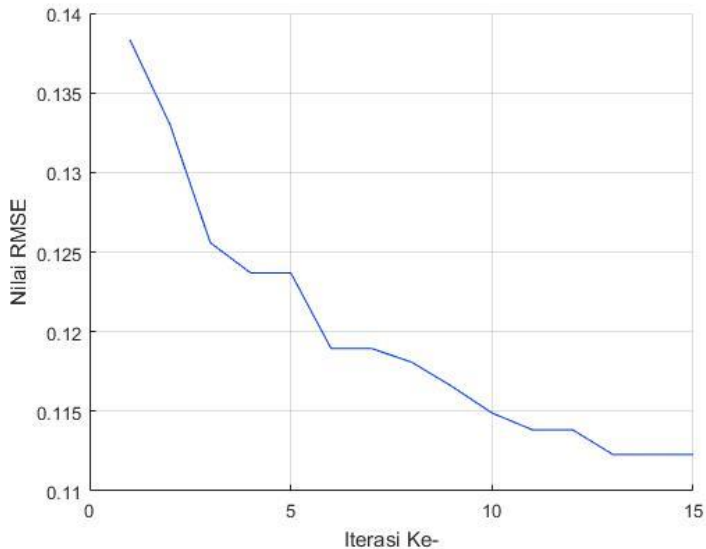
Tabel 4.10. Nilai RMSE Visibilitas dengan Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Jumlah <i>Hidden Node</i>	Metode JST	
	Nilai RMSE	
	Pelatihan	Validasi
1	0.1085	0.1006
2	0.1088	0.0989
3	0.1231	0.1346
4	0.1201	0.0944
5	0.1332	0.1
6	0.1097	0.0962
7	0.1098	0.0994
8	0.165	0.0924
9	0.1123	0.0837
10	0.1172	0.1067

Tabel 4.10. menjelaskan hasil prediksi yang berupa nilai RMSE. Nilai RMSE terendah yaitu sebesar 0,0837 terdapat pada arsitektur jaringan saraf tiruan dengan jumlah *hidden node* sebanyak 9 *node*. Jika dibandingkan dengan metode tanpa optimisasi, nilai RMSE rata-rata yang dihasilkan lebih besar, tetapi pada 9 *hidden node* memiliki nilai RMSE yang lebih kecil. Kesimpulan yang dapat diambil yaitu metode gabungan jaringan saraf tiruan dan *particle swarm optimizaton* dapat digunakan untuk melakukan prediksi cuaca dan dapat menghasilkan bobot yang optimum, bahkan dapat memperoleh hasil yang lebih baik.

4.3.5 Performansi Pelatihan Visibilitas Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Prediksi visibilitas menggunakan metode gabungan juga digunakan algoritma pelatihan *particle swarm optimization*, dengan jumlah iterasi yang digunakan sebanyak 15 iterasi. Grafik performansi yang dapat dilihat pada gambar dibawah ini menunjukkan perubahan nilai RMSE pelatihan data visibilitas pada setiap iterasi.



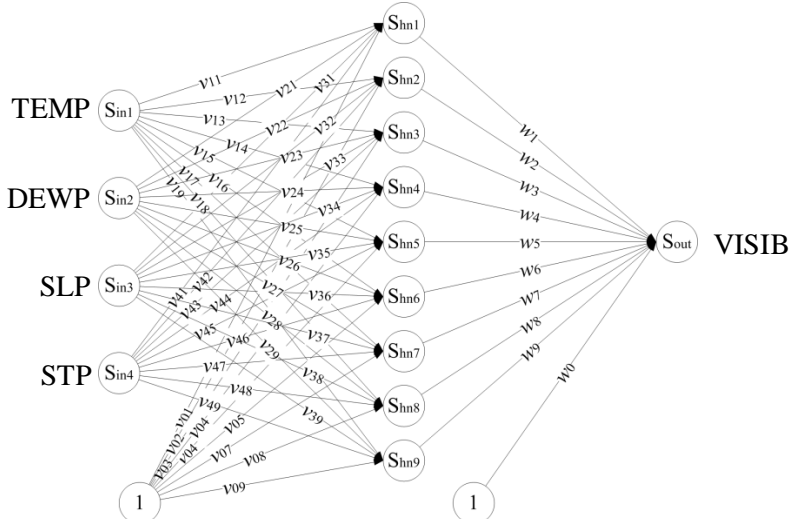
Gambar 4.18. Grafik performansi pelatihan JST-PSO untuk data visibilitas

Grafik pada gambar diatas menunjukkan bahwa nilai bobot yang terdapat pada jaringan saraf tiruan dapat diperbaharui dan menghasilkan nilai RMSE yang semakin kecil seiring dengan bertambahnya iterasi. Kesimpulan yang dapat diambil dari hal tersebut yaitu *particle swarm optimization* dapat digunakan untuk melakukan pencarian bobot optimum pada jaringan saraf tiruan.

4.3.6 Jumlah *Hidden Node* untuk Prediksi Visibilitas (Jarak Pandang) Menggunakan Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Variabel masukan yang tepat untuk melakukan prediksi visibilitas telah diperoleh dari hasil pengujian data. Jumlah *hidden node* yang menghasilkan nilai RMSE terkecil juga telah diperoleh dari percobaan sebelumnya. Langkah berikutnya yang dilakukan juga tetap sama dengan percobaan pada prediksi kecepatan angin menggunakan metode gabungan. Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah.

Masukan jaringan saraf tiruan yang diperoleh untuk prediksi visibilitas berjumlah 4 variabel yaitu TEMP, DEWP, SLP, dan STP. Selain itu, untuk jumlah *hidden node* yang digunakan diambil dari hasil pengujian jumlah *hidden node* berjumlah 9 *node*. Arsitektur jaringan saraf tiruan untuk prediksi kecepatan angin dengan menggunakan metode gabungan jaringan saraf tiruan dan *particle swarm optimization* dapat dilihat pada gambar dibawah ini.



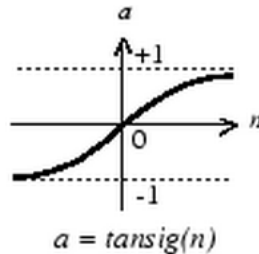
Gambar 4.19. Arsitektur JST-PSO untuk prediksi visibilitas

4.4 Parameter Arsitektur Jaringan Saraf Tiruan dari Hasil Percobaan

4.4.1 Fungsi Aktivasi

Fungsi aktivasi digunakan untuk menghasilkan nilai keluaran dari setiap node yang dilewati. Terdapat beberapa jenis fungsi aktivasi yang paling sering digunakan pada jaringan saraf tiruan yaitu sigmoid logaritmik (*logsig*), sigmoid tangen (*tansig*), dan linear (*purelin*). Fungsi aktivasi sigmoid logaritmik dan sigmoid tangen dapat meminimalisir kapasitas komputasi untuk pelatihan, sedangkan fungsi aktivasi linear lebih banyak digunakan pada *output node* (Montesdeoca-Contreras, Zambrano-Abad, Morales-Garcia, & Ávila-Campoverde, 2014).

Fungsi aktivasi pada penelitian tugas akhir ini, diaplikasikan pada setiap *hidden node* ke-*j* dengan simbol S_{hj} dan *ouput node* dengan simbol S_{out} . Fungsi aktivasi yang digunakan adalah sigmoid tangen atau dapat disebut juga dengan *binnary sigmoid* seperti pada gambar dibawah ini.



Gambar 4.20. Fungsi aktivasi sigmoid tangen (*binnary sigmoid*) pada JST

Sigmoid tangen digunakan pada penelitian ini, karena pada penelitian yang sudah ada telah dilakukan perbandingan antara kombinasi fungsi aktivasi antara sigmoid logaritmik (*logsig*), sigmoid tangen (*tansig*), dan linear (*purelin*) dengan hasil kombinasi sigmoid tangen pada *hidden node* dan *output node* memiliki hasil yang terbaik (Choudhary, Rishi, Ahlawat, & Dhaka, 2010). Hal tersebut juga telah dibuktikan dengan hasil pada penelitian tugas akhir ini yang mana semua nilai RMSE yang

dihasilkan pada prediksi kecepatan angin dan visibilitas dengan dua metode yang berbeda menghasilkan nilai yang mendekati nilai nol yang menunjukkan akurasi yang baik.

4.4.2 Faktor Bobot Optimum

Algoritma pelatihan yang digunakan pada prediksi cuaca dengan metode jaringan saraf tiruan adalah algoritma Levenberg-Marquardt, sedangkan pada metode gabungan menggunakan algoritma *particle swarm optimization*. Algoritma pelatihan yang digunakan pada metode gabungan dapat menghasilkan bobot yang optimum dengan nilai RMSE yang tidak jauh berbeda dengan metode jaringan saraf tiruan.

Faktor bobot dapat disimbolkan dengan huruf v_{ij} untuk faktor bobot yang masuk ke *hidden node* dan w_j untuk faktor bobot yang masuk ke *output node*. Huruf i dan j secara berurutan menunjukkan penamaan untuk *input node* dan *hidden node*. Sehingga, v_{ij} menunjukkan faktor bobot yang berasal dari *input node* ke- i menuju *hidden node* ke- j , sedangkan w_j menunjukkan faktor bobot yang berasal dari *hidden node* ke- j .

Tabel 4.11. Faktor Bobot Optimum dari Metode JST untuk Prediksi Kecepatan Angin

Faktor Bobot	Nilai Bobot	Faktor Bobot	Nilai Bobot
v_{11}	0.711991	v_{13}	-1.86042
v_{21}	-0.1525	v_{23}	3.868257
v_{31}	0.067148	v_{33}	-4.17592
v_{41}	-2.13777	v_{43}	-2.3687
v_{01}	-1.92128	v_{03}	-1.54903
v_{12}	2.329846	w_1	-0.64905
v_{22}	-9.36848	w_2	-0.11176
v_{32}	15.20596	w_3	-0.25251
v_{42}	-0.99033	w_0	-0.3912
v_{02}	2.048487		

Tabel 4.12. Faktor Bobot Optimum dari Metode JST untuk Prediksi Visibilitas

Faktor Bobot	Nilai Bobot	Faktor Bobot	Nilai Bobot
v_{11}	-3.21139	v_{43}	-51.8384
v_{21}	10.63658	v_{03}	-0.50994
v_{31}	7.823348	v_{14}	0.334809
v_{41}	5.030405	v_{24}	-2.34778
v_{01}	5.207337	v_{34}	-3.0751
v_{12}	1.444098	v_{44}	2.067338
v_{22}	-2.45193	v_{04}	2.38889
v_{32}	-3.31605	w_1	-0.09276
v_{42}	2.588791	w_2	0.693085
v_{02}	1.877563	w_3	0.137735
v_{13}	-1.18849	w_4	-0.98917
v_{23}	0.503901	w_0	0.935819
v_{33}	50.91957		

Nilai faktor bobot yang dihasilkan dari prediksi visibilitas dengan metode jaringan saraf tiruan dengan algoritma pelatihan Levenberg-Marquardt dapat dilihat pada tabel 4.12. Bobot yang dihasilkan pada prediksi kecepatan angin menggunakan metode jaringan saraf tiruan berjumlah 25 disesuaikan dengan banyaknya *hidden node* yang digunakan yaitu 4 *node*.

Nilai faktor bobot yang dihasilkan dari prediksi kecepatan angin dengan metode gabungan JST-PSO dengan algoritma pelatihan yang digunakan adalah *particle swarm optimization* dapat dilihat pada tabel 4.13. Bobot yang dihasilkan pada prediksi kecepatan angin menggunakan metode jaringan saraf tiruan berjumlah 37 disesuaikan dengan banyaknya *hidden node* yang digunakan yaitu 6 *node*.

Tabel 4.13. Faktor Bobot Optimum dari Metode JST-PSO untuk Prediksi Kecepatan Angin

Faktor Bobot	Nilai Bobot	Faktor Bobot	Nilai Bobot
V ₁₁	-0.17576	V ₀₄	-0.0816
V ₂₁	0.156778	V ₁₅	-0.24102
V ₃₁	-0.03662	V ₂₅	0.167449
V ₄₁	-1	V ₃₅	0.059084
V ₀₁	0.334296	V ₄₅	0.255907
V ₁₂	0.430238	V ₀₅	-0.18747
V ₂₂	-0.11797	V ₁₆	0.105669
V ₃₂	-0.62101	V ₂₆	0.170089
V ₄₂	-0.62948	V ₃₆	0.581147
V ₀₂	-0.06366	V ₄₆	0.017914
V ₁₃	-0.96789	V ₀₆	-0.42125
V ₂₃	-0.70496	W ₁	-0.89092
V ₃₃	-0.17645	W ₂	-0.62928
V ₄₃	-0.37286	W ₃	0.007755
V ₀₃	-0.08432	W ₄	-0.56823
V ₁₄	-0.23523	W ₅	0.250501
V ₂₄	0.81948	W ₆	-0.25091
V ₃₄	-0.61513	W ₀	0.215929
V ₄₄	-0.31773		

Nilai faktor bobot yang dihasilkan dari prediksi visibilitas dengan metode gabungan JST-PSO dengan algoritma pelatihan yang digunakan adalah *particle swarm optimization* dapat dilihat pada tabel 4.14. Bobot yang dihasilkan pada prediksi kecepatan angin menggunakan metode jaringan saraf tiruan berjumlah 55 disesuaikan dengan banyaknya *hidden node* yang digunakan yaitu 9 *node*.

Tabel 4.14. Faktor Bobot Optimum dari Metode JST-PSO untuk Prediksi Visibilitas

Faktor Bobot	Nilai Bobot	Faktor Bobot	Nilai Bobot
v ₁₁	0.049321	v ₄₆	-0.01964
v ₂₁	-0.59731	v ₀₆	-0.38404
v ₃₁	0.16221	v ₁₇	0.092386
v ₄₁	-0.24359	v ₂₇	-0.74793
v ₀₁	-0.29786	v ₃₇	-0.03888
v ₁₂	-0.26889	v ₄₇	0.435588
v ₂₂	-0.46916	v ₀₇	0.5467
v ₃₂	-0.02016	v ₁₈	0.022609
v ₄₂	-0.61217	v ₂₈	0.572439
v ₀₂	-0.12443	v ₃₈	0.23365
v ₁₃	0.276376	v ₄₈	-0.18941
v ₂₃	0.208863	v ₀₈	0.34189
v ₃₃	0.39076	v ₁₉	0.591929
v ₄₃	-0.10707	v ₂₉	-0.09969
v ₀₃	0.246227	v ₃₉	-0.05496
v ₁₄	0.289089	v ₄₉	-0.32435
v ₂₄	-0.28709	v ₀₉	0.168814
v ₃₄	-0.18207	w ₁	-0.245
v ₄₄	0.458019	w ₂	0.300996
v ₀₄	0.176385	w ₃	0.11044
v ₁₅	0.556606	w ₄	0.02428
v ₂₅	-0.55897	w ₅	0.511489
v ₃₅	0.240537	w ₆	-0.07181
v ₄₅	0.491824	w ₇	0.117367
v ₀₅	0.090763	w ₈	-0.20027
v ₁₆	-0.29964	w ₉	0.883987
v ₂₆	-0.27959	w ₀	0.523743
v ₃₆	0.307807		

4.5 Perbandingan Hasil Antara Kedua Metode

Percobaan prediksi cuaca telah dilakukan menggunakan metode jaringan saraf tiruan dan metode gabungan jaringan saraf tiruan dan *particle swarm optimization* dengan variasi *hidden node*. Hasil terbaik yang telah diperoleh dan dapat dibandingkan yaitu nilai RMSE prediksi terkecil, jumlah *hidden node* terbaik, dan iterasi pada saat diperolehnya nilai RMSE tersebut.

Tabel 4.15. Perbandingan hasil terbaik antara metode JST dan metode gabungan JST-PSO

Variabel		JST	JST-PSO
Kecepatan Angin	Target Nilai RMSE Validasi	0,0721	0,0721
	Jumlah <i>Hidden Node</i>	6	6
	Diperoleh pada Epoch Ke-	3	11
Visibilitas	Target Nilai RMSE Validasi	0,0837	0,0837
	Jumlah <i>Hidden Node</i>	9	9
	Diperoleh pada Epoch Ke-	Tidak Tercapai	13

Tabel 4.15. menjelaskan perbandingan performa antara kedua metode tersebut yang dibagi menjadi dua prediksi yaitu prediksi kecepatan angin dan visibilitas, dan terdapat dua metode yaitu metode jaringan saraf tiruan (JST) dan metode gabungan jaringan saraf tiruan dan *particle swarm optimization* (JST-PSO). Parameter yang dapat dibandingkan antara kedua metode prediksi tersebut yaitu jumlah iterasi untuk memperoleh nilai RMSE yang sama dengan menggunakan jumlah *hidden node* yang sama.

Jumlah *hidden node* yang digunakan pada prediksi kecepatan angin adalah 6 *node* dengan nilai RMSE yang digunakan sebagai target adalah 0,0721. Metode JST yang digunakan untuk melakukan prediksi kecepatan angin dapat menghasilkan nilai

RMSE 0,0721 pada epoch ke-3, sedangkan metode JST-PSO pada epoch ke-11. Kesimpulan yang dapat diambil dari hal ini yaitu metode JST-PSO memiliki performa yang kurang baik dalam melakukan prediksi kecepatan angin, tetapi tetap dapat menghasilkan bobot yang optimum karena nilai RMSE yang dihasilkan mendekati nol.

Jumlah *hidden node* yang digunakan pada prediksi visibilitas yaitu berjumlah 9 *node* dengan nilai RMSE yang digunakan sebagai target sebesar 0,0837. Metode JST-PSO pada percobaan ini dapat menghasilkan nilai RMSE sebesar 0,0837 pada epoch ke-13, sedangkan metode JST tidak dapat mencapai nilai RMSE ter-sebut. Kesimpulan yang dapat diambil dari percobaan ini yaitu metode JST-PSO memiliki performa yang lebih baik dalam melakukan prediksi visibilitas, dan juga dapat menghasilkan bobot yang optimum.

4.6 Keselamatan Penerbangan Menggunakan Metode Jaringan Saraf Tiruan dan *Particle Swarm Optimization*

Keselamatan penerbangan dapat ditingkatkan dengan melakukan prediksi cuaca, terutama dapat dilakukan untuk pesawat mendarat dan lepas landas. Pada penelitian ini dilakukan simulasi dari prediksi dengan menggunakan data validasi sebanyak 456 set data, yang dimulai dari data cuaca pada tanggal 1 Oktober 2015 hingga 31 Desember 2016. Pada rentang waktu tersebut telah dilakukan simulasi dan validasi prediksi cuaca, dan kemudian dapat dilihat pada tanggal berapa saja nilai kecepatan angin dan visibilitas yang dapat membahayakan penerbangan.

Besar kecepatan angin yang dapat membahayakan adalah sebesar 10 knot dengan arah angin 90° dari arah datangnya pesawat saat lepas landas ataupun mendarat. Penelitian yang dilakukan hanya dalam batasan kecepatan angin, sehingga yang perlu diperhatikan adalah pada tanggal berapa kecepatan angin yang melewati batas aman. Besar kecepatan angin yang melewati batas aman antara tanggal 1 Oktober 2015 hingga 31 Desember 2016 dapat dilihat pada tabel dibawah ini.

Tabel 4.16. Besar Kecepatan Angin yang Melewati Batas Aman

Tanggal	Kecepatan Angin (Knot)	Visibilitas (m)
22 Desember 2015	13,13142015	7144,86
24 Mei 2016	13,45836453	7711,27
25 Mei 2016	13,48272259	7834,20
31 Juli 2016	12,88043808	7081,25

Jarak aman yang dianjurkan untuk visibilitas dalam melakukan pendaratan pesawat pada Bandar Udara Internasional Juanda adalah sebesar 1600 meter atau kurang lebih 1 mil. Nilai prediksi visibilitas yang diperoleh dari hasil simulasi tidak ada yang dibawah 1 mil. Kesimpulan dari hasil tersebut yaitu dikatakan antara tanggal 1 Oktober 2015 hingga 31 Desember 2016, tidak terdapat faktor-faktor yang menyebabkan buruknya visibilitas pada Bandar Udara Internasional Juanda.

Halaman ini sengaja dikosongkan

BAB V

KESIMPULAN

5.1 Kesimpulan

Terdapat beberapa kesimpulan yang diperoleh dari penelitian tugas akhir ini, yaitu sebagai berikut:

- a. Metode gabungan JST-PSO dapat meningkatkan akurasi pada prediksi visibilitas menggunakan 9 *hidden node* dengan nilai RMSE yang dihasilkan sebesar 0,0837 dibandingkan dengan metode JST dengan nilai RMSE sebesar 0,08606.
- b. Metode *Particle Swarm Optimization* dapat menghasilkan bobot yang optimum pada Jaringan Saraf Tiruan, dilihat dari nilai RMSE yang dihasilkan metode gabungan JST-PSO mendekati nilai nol. Nilai RMSE terbaik menggunakan metode gabungan JST-PSO untuk prediksi kecepatan angin sebesar 0,0721 dengan menggunakan 6 *hidden node*, dan untuk prediksi visibilitas sebesar 0,0837 dengan menggunakan 9 *hidden node*.
- c. Metode gabungan JST-PSO dapat digunakan untuk meningkatkan keselamatan penerbangan dilihat dari hasil simulasi prediksi yang menghasilkan kecepatan angin yang melewati batas aman pada tanggal 13 Desember 2015 sebesar 13,13 knot, 24 Mei 2016 sebesar 13,45 knot, 25 Mei 2016 sebesar 13,48 knot, dan 31 Juli 2016 sebesar 12,88 knot.

5.2 Saran

Saran yang dapat diajukan terkait dengan penelitian tugas akhir ini, antara lain:

- a. Percobaan yang dapat dilakukan pada penelitian selanjutnya yaitu dengan menggunakan metode optimisasi selain *Particle Swarm Optimization* untuk melihat apakah semua metode optimisasi dapat memberikan hasil yang lebih baik dibandingkan tanpa menggunakan metode optimisasi.
- b. Sebaiknya dilakukan juga penelitian dengan membandingkan hasil prediksi pada penelitian yang telah dilakukan dengan metode prediksi cuaca yang sedang digunakan pada bagian

prakiraan cuaca Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Juanda, Surabaya.

- c. Selain itu, pada penelitian sejenis dapat juga ditambahkan penelitian untuk melakukan prediksi cuaca dimasa depan, dimulai dari cuaca beberapa jam kedepan hingga beberapa tahun kedepan.

DAFTAR PUSTAKA

- Ahrens, C. D. (2003). *Meteorology Today : An Introduction to Weather, Climate, and The Environment* (Seventh ed.). (K. Dodson, Ed.) Pacific Grove, United States of America: Thomson Learning, Inc.
- Allaby, M. (2007). *Encyclopedia of Weather and Climate* (Revised ed.). New York, United States of America: Facts On File, Inc.
- Amaliah, B., & Okafor, A. (2011). Pemilihan Warna Lipstik Berdasarkan Informasi Usia Dan Warna Kulit Dengan Menggunakan Metoda Artificial Neural Network. *Seminar Nasional Teknologi Informasi & Komunikasi Terapan*.
- Atiliani, A. (2013). *Pelatihan Jaringan Syaraf Tiruan Multilayer Perceptron Menggunakan Genetic Algorithm Levenberg Marquardt*. Surakarta: Perpustakaan Universitas Sebelas Maret.
- BMKG Juanda. (2017, January 26 & 29). Sistem Pemantauan dan Prediksi Cuaca pada Badan Meteorologi, Klimatologi, & Geofisika. (E. F. Hartantyo, & D. A. Rafika, Interviewers)
- Choudhary, A., Rishi, R., Ahlawat, S., & Dhaka, V. S. (2010). Performance Analysis of Feed Forward MLP with various Activation Functions for Handwritten Numerals Recognition. *The 2nd International Conference on Computer and Automation Engineering (ICCAE)*. 5, pp. 852-856. Singapore: IEEE.
- Clerc, M. (2006). *Particle Swarm Optimization*. London, United Kingdom: ISTE Ltd.
- Fausett, L. V. (1994). *Fundamental of Neural Networks: Architectures, Algorithms, and Applications* (Illustrated ed.). Prentice-Hall.
- Ghaffari, A., Abdollahi, H., Khoshayand, M., Bozchalooi, I. S., Dadgar, A., & Rafiee-Tehrani, M. (2008). Performance comparison of neural network training algorithms in

- modeling of bimodal drug delivery. *International Journal of Pharmaceutics*, (pp. 126-138).
- Kementrian Perhubungan Republik Indonesia. (2017, January 28). *Bandar Udara : Semua Propinsi, Semua Kategori*. Retrieved January 29, 2017, from Direktorat Jenderal Perhubungan Udara:
<http://hubud.dephub.go.id/?id=bandara/index/filter:category,0>
- Keshavarzi, A., & Sarmadian, F. (2010). Comparison of artificial neural network and multivariate regression methods in prediction of soil cation exchange capacity (Case study: Ziara region). *Desert*, 167-174.
- Kulesa, G. (2003). *The Potential Impacts of Climate Change on Transportation*. Retrieved December 28, 2016, from DOT Transportation and Climate Change Clearinghouse:
<https://climate.dot.gov/documents/workshop1002/kulesa.pdf>
- Mahapatra, P. (1999). *Aviation Weather Surveillance Systems - Advanced radar and surface sensors for flight safety and air traffic management*. Stevenage, Herts, United Kingdom: The Institution of Electrical Engineers.
- Mislan, Haviluddin, Hardwinarto, S., Sumaryono, & Marlon, A. (2015). Rainfall Monthly Prediction Based on Artificial Neural Network: A Case Study in Tenggarong Station, East Kalimantan - Indonesia. *International Conference on Computer Science and Computational Intelligence (ICCSCI 2015)*. *Procedia Computer Science* 59, pp. 142-151. Elsevier.
- Montesdeoca-Contreras, J. C., Zambrano-Abad, J. C., Morales-Garcia, J. A., & Ávila-Campoverde, R. S. (2014). Virtual Speed Sensor for DC Motor using Back-Propagation Artificial Neural Networks. *IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*. Ixtapa, Mexico: IEEE.

- National Climatic Data Center, U.S. Department of Commerce. (n.d.). *Climate Data Online*. Retrieved April 27, 2017, from www7.ncdc.noaa.gov/CDO/dataproduct
- Nayar, R., Patheja, P. S., & Wao, A. A. (2012). An Artificial Neural Network Model for Weather Forecasting in Bhopal. *International Conference on Advances In Engineering, Science And Management (ICAESM 2012)* (pp. 747-749). IEEE.
- Nugraha, H. G., & SN, A. (2014). Optimasi Bobot Jaringan Syaraf Tiruan Menggunakan Particle Swarm Optimization. *Indonesian Journal of Computing and Cybernetics Systems (IJCCS)*. 8, pp. 25-36. Indonesian Computer, Electronics, and Instrumentation Support Society (IndoCEISS).
- OAG Aviation Worldwide Limited. (2017, January). *Punctuality League 2016 - Most Punctual Airlines and Airports in 2016*. Retrieved January 15, 2017, from The Largest Airlines Schedule and Flight Status Database in The World:
https://www.oag.com/hubfs/Free_Reports/Punctuality_League/2016/PunctualityReport2016.pdf?__hssc=238362719.3.1483539331973&__hstc=238362719.d8cfccb617dc4f03e1419d5e4346e737.1483539331970.1483539331970.1483539331970.1&__hsfp=1047458973&hsCtaTracking=668020f
- Priyatno, D. (2009). *5 Jam Belajar Olah Data dengan SPSS 17*. (J. Widiyatmoko, Ed.) Yogyakarta, Indonesia: C.V ANDI OFFSET.
- Rao, S. S. (2009). *Engineering Optimization: Theory and Practice* (Fourth ed.). Hoboken, New Jersey, Canada: John Wiley & Sons, Inc.
- Riadi, J., & Nurmahaludin. (2012). Aplikasi Jaringan Syaraf Tiruan Multi Layer Perceptron Pada Aplikasi Prakiraan Cuaca. *Jurnal POROS TEKNIK*, 4(2), 71-76.
- Santoso, S. (2010). *Statistik Multivariat*. Jakarta, Indonesia: PT Elex Media Komputindo.

- Sarwono, J. (2006). *Analisis Data Penelitian Menggunakan SPSS 13*. (D. Hardjono, Ed.) Yogyakarta, Indonesia: C.V ANDI OFFSET.
- Sugiyono. (2017). *Statistika Untuk Penelitian*. Bandung, Jawa Barat, Indonesia: Alfabeta.
- Tjasjono, B. (1999). *Klimatologi Umum*. Bandung, Jawa Barat, Indonesia: Institut Teknologi Bandung(ITB).
- Trihendradi, C. (2012). *Step by Step SPSS 20 Analisis Data Statistik*. (F. S. Suyantoro, Ed.) Yogyakarta, Indonesia: C.V ANDI OFFSET.
- World Bank. (2017, January 3rd). *Data : Air Transport, Passengers Carried, Indonesia*. Retrieved January 5th, 2017, from The World Bank:
<http://data.worldbank.org/indicator/IS.AIR.PSGR?locations=ID>
- Zhao, H., Jin, L., & Huang, X. (2010). A Prediction of Monthly Precipitation Model Based on PSO-ANN and Its Applications. *Third International Joint Conference on Computational Science and Optimization* (pp. 476-479). IEEE.
- Zhao, H., Jin, L., Huang, Y., & Huang, X. (2012). An application of ensemble prediction for typhoon intensity based on MDS and PSO-ANN. *Fifth International Joint Conference on Computational Sciences and Optimization* (pp. 885-888). IEEE.

LAMPIRAN

Lampiran 1. Pengujian Data

– Pengujian Data Cuaca yang Hilang

Analisa data cuaca yang hilang dilakukan untuk melihat apakah jumlah data yang hilang mengganggu hasil prediksi dan juga untuk melihat data yang hilang bersifat acak atau tidak. Data yang hilang sebenarnya tidak akan berpengaruh pada keseluruhan data, jika berjumlah sedikit yaitu sekitar 1% dari jumlah keseluruhan data (Santoso, 2010). Jumlah data yang hilang pada data variabel cuaca dapat dilihat pada tabel dibawah ini.

Tabel Lampiran 1. Jumlah Data Variabel Cuaca yang Hilang

Variabel	Jumlah Data	Mean	Standar Deviasi	<i>Missing Data</i>	
				<i>Count</i>	<i>Percent</i>
TEMP	1821	82.628	2.1147	6	0.3
DEWP	1821	73.879	2.7665	6	0.3
SLP	1821	1010.117	1.6085	6	0.3
STP	1821	1009.622	1.6410	6	0.3
MXSPD	1821	10.664	3.6380	6	0.3
MAX T.	1821	89.696	2.3533	6	0.3
MIN T.	1821	76.014	2.4103	6	0.3
VISIB	1821	4.866	.6863	6	0.3
WDSP	1821	5.737	1.9181	6	0.3

Dari tabel tersebut dapat dilihat bahwa semua variabel memiliki data yang hilang dengan jumlah yang sama yaitu 6 data, dan dengan persentase 0.3%. Persentase tersebut lebih kecil dari batas persentase data hilang yang diperbolehkan yaitu sebesar 1%. Sehingga dapat disimpulkan bahwa data tersebut dapat digunakan.

Kemudian dari data tersebut di analisa apakah pola data yang hilang bersifat acak atau tidak. Pola data yang hilang dapat dilihat pada tabel 2.

Tabel Lampiran 2. Pola Data yang Hilang Pada Tiap Variabel Cuaca

Data Ke-	Jumlah		Pola Data yang Hilang (<i>Missing Data</i>)								
	N	%	TEMP	DEWP	SLP	STP	MXSPD	MAX T.	MIN T.	VISIB	WDSP
564	9	100	×	×	×	×	×	×	×	×	×
617	9	100	×	×	×	×	×	×	×	×	×
618	9	100	×	×	×	×	×	×	×	×	×
925	9	100	×	×	×	×	×	×	×	×	×
1544	9	100	×	×	×	×	×	×	×	×	×
1777	9	100	×	×	×	×	×	×	×	×	×
*Simbol × menunjukkan pada data seberapa dan variabel apa saja yang hilang											

Dari tabel tersebut dapat dilihat bahwa pola data yang hilang bersifat tidak acak, karena data yang hilang terdapat pada 6 set data dengan semua variabel cuaca yang tidak terdapat nilainya atau hilang. Untuk mengatasi hal ini, dapat dilakukan dengan membuang atau menghapus baris (data ke-) ataupun kolom (variabel) yang mengandung data yang hilang (Santoso, 2010). Dalam hal ini yang akan dihapus adalah baris yang mengandung data yang hilang.

– Pengujian Data *Outlier* Cuaca

Pengujian terhadap data yang ekstrem atau *outlier* dapat dilakukan dengan membuat nilai z atau nilai standardisasi data. Pada prinsipnya, nilai variabel cuaca yang sebenarnya diubah dalam bentuk nilai z , kemudian dari hasil tersebut dapat ditafsirkan. Sebuah data dapat disebut *outlier*, jika memiliki nilai z yang lebih besar dari nilai 2.5 dan lebih kecil dari nilai -2.5. Kemudian dari nilai z tersebut dapat disimpulkan beberapa nilai variabel cuaca yang outlier sebagai berikut.

Tabel Lampiran 3. Hasil Pengujian Data *Outlier* Variabel Suhu Udara

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
TEMP	1423	88.6
	319	88.5
	<i>Highest</i> 1418	88.3
	1424	88.0
	314	87.9
	21	76.0
	1193	76.5
	<i>Lowest</i> 777	76.8
	1758	77.0
	1076	77.3

Tabel Lampiran 4. Hasil Pengujian Data *Outlier* Variabel Titik Embun

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
DEWP	470	78.8
	1531	78.6
	<i>Highest</i> 1591	78.6
	453	78.5
	1559	78.5
	590	63.8
	228	64.9
	<i>Lowest</i> 1000	65.2
	1310	65.4
	589	65.6

Tabel Lampiran 5. Hasil Pengujian Data *Outlier* Variabel Tekanan Udara Diatas Permukaan Laut

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
SLP	1275	1014.8
	1276	1014.8
	<i>Highest</i> 1344	1014.4
	1303	1014.3
	1343	1014.3
	68	1004.4
	376	1004.7
	<i>Lowest</i> 375	1004.9
	1818	1005.0
	69	1005.1

Tabel Lampiran 6. Hasil Pengujian Data *Outlier* Variabel Tekanan Udara Pada Stasiun Cuaca

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
STP	1275	1014.5
	1276	1014.4
	<i>Highest</i> 1343	1014.0
	1344	1014.0
	1372	1014.0
	68	1003.8
	376	1004.0
	<i>Lowest</i> 375	1004.3
	69	1004.4
	24	1004.6

Tabel Lampiran 7. Hasil Pengujian Data *Outlier* Variabel Kecepatan Udara Maksimum

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
MXSPD	702	48.6
	1606	42.9
	<i>Highest</i> 1607	42.9
	727	42.7
	1172	40.0 ^d
	1795	2.9
	1465	4.1
	<i>Lowest</i> 1219	4.1
	720	4.1
	524	4.1 ^e

Tabel Lampiran 8. Hasil Pengujian Data *Outlier* Variabel Suhu Udara Maksimum

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
MAX T.	1041	96.1
	1423	96.1
	<i>Highest</i> 320	95.7
	1401	95.7
	1424	95.7 ^f
	752	79.3
	550	80.1
	<i>Lowest</i> 513	80.2
	135	80.6
	534	81.0

Tabel Lampiran 9. Hasil Pengujian Data *Outlier* Variabel Suhu Udara Minimum

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
MIN T.	105	81.3
	320	81.3
	<i>Highest</i> 657	81.3
	1607	81.3
	1723	81.3
	21	57.7
	1335	63.9
	<i>Lowest</i> 591	67.5
	590	67.5
	1771	68.4 ^g

Tabel Lampiran 10. Hasil Pengujian Data *Outlier* Variabel Vi-sibilitas

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
VISIB	35	6.2
	80	6.2
	<i>Highest</i> 93	6.2
	100	6.2
	105	6.2 ^h
	776	1.5
	163	2.0
	<i>Lowest</i> 695	2.5
	396	2.6
	717	2.7

Tabel Lampiran 11. Hasil Pengujian Data *Outlier* Variabel Kecepatan Angin

<i>Variable</i>	<i>Case Number</i>	<i>Value</i>
WDSP	1290	14.7
	1372	14.2
	<i>Highest</i> 1607	13.7
	948	13.6
	26	13.1
	1795	0.4
	17	0.5
	<i>Lowest</i> 534	0.9
	163	0.9
	398	1.2

Dari tabel diatas dapat dilihat bahwa terdapat banyak data yang bernilai ekstrem atau *outlier* pada data variabel cuaca yang telah diperoleh. Data *outlier* dapat ditangani dengan menghilangkan data karena dianggap tidak mencerminkan sebaran data yang sesungguhnya, atau tetap dipertahankan dan tidak perlu dihilangkan karena mungkin memang terdapat data *outlier* seperti itu.

Dalam kasus prediksi cuaca, data outlier akan tetap dipertahankan karena memang terdapat data variabel cuaca yang bernilai ekstrem dan karena memang dibutuhkan seluruh data yang sebenarnya secara lengkap baik bernilai normal ataupun ekstrem.

– Pengujian Normalitas Data Cuaca

Pengujian normalitas data dapat dilakukan dengan menggunakan dua metode, yaitu metode kolmogorov-smirnov dan metode saphiro-wilk. Kedua metode tersebut dapat digunakan pada *software IBM SPSS Statistics 24* dengan hasil seperti pada tabel.

Tabel Lampiran 12. Hasil Pengujian Normalitas Data Cuaca

Variable	Kolmogorov-Smirnov			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
TEMP	0.036	1821	1.310E-05	0.996	1821	5.372E-05
DEWP	0.147	1821	1.538E-108	0.920	1821	4.149E-30
SLP	0.032	1821	1.958E-04	0.997	1821	6.514E-04
STP	0.035	1821	2.345E-05	0.996	1821	2.193E-04
MXSPD	0.138	1821	4.328E-96	0.810	1821	6.330E-42
MAX T.	0.042	1821	7.970E-08	0.992	1821	1.602E-08
MIN T.	0.095	1821	2.847E-44	0.957	1821	1.039E-22
VISIB	0.071	1821	3.133E-24	0.984	1821	3.055E-13
WDSP	0.044	1821	1.455E-08	0.979	1821	1.195E-15

Data dapat dikatakan berdistribusi normal jika memiliki angka signifikansi (SIG) lebih besar dari 0.05, selain itu maka data dapat dianggap tidak berdistribusi normal (Santoso, 2010). Dapat dilihat pada tabel 4.12 bahwa seluruh angka signifikansi bernilai lebih kecil dari 0.05 sehingga dapat diperoleh kesimpulan bahwa seluruh variabel data cuaca tidak berdistribusi normal.

Data yang tidak berdistribusi normal dapat ditangani dengan beberapa cara yaitu menambah data, menghilangkan data penyebab tidak berdistribusi normal, melakukan transformasi data, dan diterima apa adanya. Dalam kasus prediksi cuaca data dapat diterima apa adanya.

Lampiran 2. Kode Program Prediksi Cuaca Metode JST

```

close all;
clear all;
clc
disp(' PELATIHAN NEURAL NETWORK ')

A = xlsread('D:\Undergraduate Thesis\Data
Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx','Wind','H2:L1822');

rv = 1366;           % row validation start
re = 1821;           % row end
io = 4;              % jumlah input
hidden = 10;         % jumlah hidden node

ut = A(1:rv-1,1:4)'; %input di transpose
yt = A(1:rv-1,5)';   % output di transpose

uv = A(rv:re,1:4)'; %input di transpose
yv = A(rv:re,5)';    %output di transpose

us = A(1:re,1:4)'; %input di transpose
ys = A(1:re,5)';    %output ditranspose

[rowv,colv] = size(uv);
[rowu,colu] = size(ut);
[rowy,coly] = size(yt);
Min = -ones(rowu,1);
Max = ones(rowu,1);
MM = [Min Max];

for i=1:rowu
    maxusa(i)=max(us(i,:));
    minusa(i)=min(us(i,:));
end

for i = 1:rowy

```

```

        maxys(i)=max(ys(i,:));
        minys(i)=min(ys(i,:));
end

minmaxus = [maxusa;minusa];
minmaxys = [maxys;minys];

for i = 1:rowy
    yt(i,:)=((2/(max(ys(i,:))-
min(ys(i,:))))*(yt(i,:)-min(ys(i,:))))-1;
    yv(i,:)=((2/(max(ys(i,:))-
min(ys(i,:))))*(yv(i,:)-min(ys(i,:))))-1;
end

for j = 1:colu
    for i=1:rowu
        ut(i,j)=((2/(maxusa(i)-
minusa(i))))*(ut(i,j)-minusa(i)))-1;
    end
end

for j = 1:colv
    for i=1:rowv
        uv(i,j)=((2/(maxusa(i)-
minusa(i))))*(uv(i,j)-minusa(i)))-1;
    end
end

ut1=ut(1,:);
ut2=ut(2,:);
ut3=ut(3,:);
ut4=ut(4,:);

yt1 = yt(1,:);

uv1=uv(1,:);
uv2=uv(2,:);
uv3=uv(3,:);

```

```

uv4=uv(4,:)';

yv1 = yv(1,:)';

% history length for MIMO identification
hist = ones(1,io);

[n_rows,n_col] = size(ut1);

% setting training data matrix
data_latih = zeros(n_rows-1,sum(hist));

for i = 1:hist(1),
    data_latih(:,i) = [zeros(hist(1)-
i,1);ut1(2:n_rows-hist(1)+i)];
end

for j = 1:hist(2),
    data_latih(:,sum(hist(1))+j) =
[zeros(hist(2)-j,1);ut2(2:n_rows-hist(2)+j)];
end

for k = 1:hist(3),
    data_latih(:,sum(hist(1:2))+k) =
[zeros(hist(3)-k,1);ut3(2:n_rows-hist(3)+k)];
end
for l = 1:hist(4),
    data_latih(:,sum(hist(1:3))+l) =
[zeros(hist(4)-l,1);ut4(2:n_rows-hist(4)+l)];
end

PHI = data_latih';

% Construction of output matrix
Y = zeros(n_rows-1,1);
Y(:,1) = yt1(2:end);

```

```

Ys = Y';

% Construction of networks structure
NetDef = [];
netdef1 = 'H';
netdef2 = 'L';
L = [netdef1;netdef2];
Data_RMSE =[];
trparms = settrain;

for x=1:hidden;
    hn = x
    close all;
    Ys = Y'
    NetDef = [NetDef L]
    netdef1 = 'H';
    netdef2 = '-';
    L = [netdef1;netdef2];

    % Construction of networks structure

    trparms = settrain;

    [W1,W2,PI_vec,yhat] =
marq_rev(NetDef,[],[],PHI,Ys,trparms);

    % RMSE calculation

    for i = 1:1

RMSE_train(i)=r_m_s_e(yhat(i,:),Ys(i,:));
    end

    for i = 1:1
        Ys(i,:)=(((max(ys(i,:))-
min(ys(i,:))))*(Ys(i,:)+1)/2)+min(ys(i,:));
    %Descaling

```

```

        Yhat(i,:)=((max(ys(i,:))-
min(ys(i,:)))*(yhat(i, :)+1)/2)+min(ys(i, :));
%Descaling
        RMSE_train_f(i) =
r_m_s_e(Ys(i,:),Yhat(i,:));

    end

%Drawing
    for i = 1

        %Denormalisasi
        R1 = Ys(i,:);
        P1 = Yhat(i,:);
        maxval = 14.7;
        minval = 0.4;

        Reall = (((maxval-minval)*(R1-
0.1))/0.8)+minval;
        Predict1 = (((maxval-minval)*(P1-
0.1))/0.8)+minval;

        figure(i)
        plot(Reall,'b-','LineWidth',1);
        hold on
        plot(Reall,'r-.','LineWidth',1);
        grid
        title('Pelatihan Data Cuaca');
        legend('Data Aktual','Data Prediksi');
        ylabel('Prediksi Kecepatan Angin
(Knot)');
        xlabel('Hari ke- (Data Latih)');
    end

    save WT_Cat NetDef W1 W2 maxys minys maxusa
minusa
    xlswrite('D:\Undergraduate Thesis\Data
Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx', W1, 'W1')

```

```

        xlswrite('D:\Undergraduate Thesis\Data
Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx', W2, 'W2')

% Tahap Validasi
disp('          VALIDASI DATA          ')

[n_rows,n_col] = size(uv1);
data_uji = zeros(n_rows-1,sum(hist));

for i = 1:hist(1),
    data_uji(:,i) = [zeros(hist(1)-
i,1);uv1(2:n_rows-hist(1)+i)];
end

for j = 1:hist(2),
    data_uji(:,sum(hist(1))+j) =
[zeros(hist(2)-j,1);uv2(2:n_rows-hist(2)+j)];
end

for k = 1:hist(3),
    data_uji(:,sum(hist(1:2))+k) =
[zeros(hist(3)-k,1);uv3(2:n_rows-hist(3)+k)];
end
for l = 1:hist(4),
    data_uji(:,sum(hist(1:3))+l) =
[zeros(hist(4)-l,1);uv4(2:n_rows-hist(4)+l)];
end

PHI_uji = data_uji';

Y_uji = zeros(n_rows-1,1);
Y_uji(:,1) = yv1(2:end);

Ys_uji = Y_uji';

```



```

[y2_uji]=marq_rev_uji(NetDef,W1,W2,PHI_uji,Ys_uj
i);

    % RMSE calculation
    for i = 1:1
        RMSE_test(i)=
r_m_s_e(Ys_uji(i,:),y2_uji(i,:))
    end

    for i = 1:1
        Ys_test(i,:)=((max(ys(i,:))-
min(ys(i,:)))*(Ys_uji(i,:)+1)/2)+min(ys(i,:));
        %Descaling
        Yhat_test(i,:)=((max(ys(i,:))-
min(ys(i,:)))*(y2_uji(i,:)+1)/2)+min(ys(i,:));
        %Descaling
        %m = Yhat_test(i,:);
        RMSE_test_f(i)=
r_m_s_e(Ys_test(i,:),Yhat_test(i,:))

        %save Yhat_test(i,:)
    end

    %Drawing
    for i = 1

        %Denormalisasi
        R2 = Ys_test(i,:);
        P2 = Yhat_test(i,:);
        maxval = 14.7;
        minval = 0.4;

        Real2 = ((maxval-minval)*(R2-
0.1))/0.8)+minval;
        Predict2 = ((maxval-minval)*(P2-
0.1))/0.8)+minval;

        figure(i+3)
    end
end

```

```

plot(Real2,'k-','LineWidth',1);
hold on
plot(Predict2,'r-','LineWidth',1);
grid
title('Validasi Data Cuaca');
legend('Aktual','Prediksi');
ylabel('Validasi Kecepatan Angin
(Knot)');
xlabel('Hari ke- (Data Validasi)');

%=====
RMSE_iterasi (i,:)= [RMSE_train RMSE_test
RMSE_train_f RMSE_test_f ];
Data_RMSE = [Data_RMSE; RMSE_iterasi];

end;

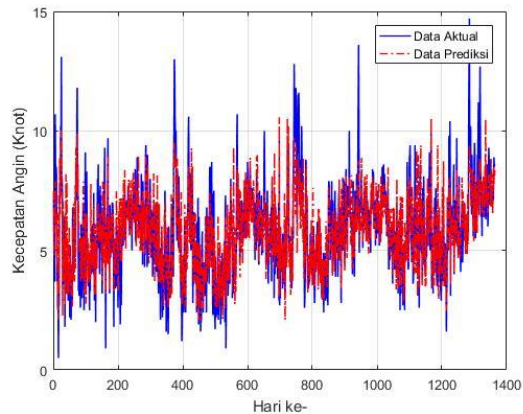
xlswrite('D:\Undergraduate Thesis\Data
Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx', RMSE_iterasi, 'RMSE iterasi 2')
xlswrite('D:\Undergraduate Thesis\Data
Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx', Data_RMSE, 'Data RMSE')
xlswrite('D:\Undergraduate Thesis\Data
Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx', Yhat_test, 'Yhat_test_transpose')

end;
save JST_Wind_All

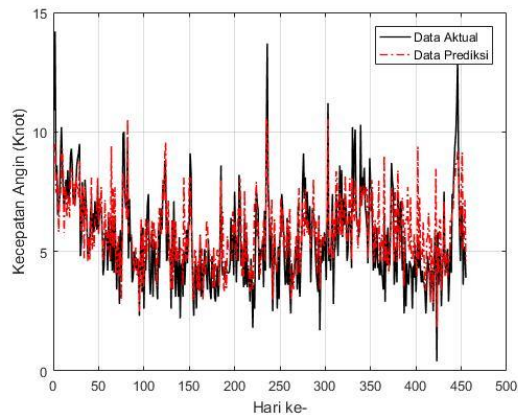
```

Lampiran 3. Grafik Pelatihan dan Validasi Data Prediksi Kecepatan Angin dengan Metode JST

- 1 *Hidden Node*

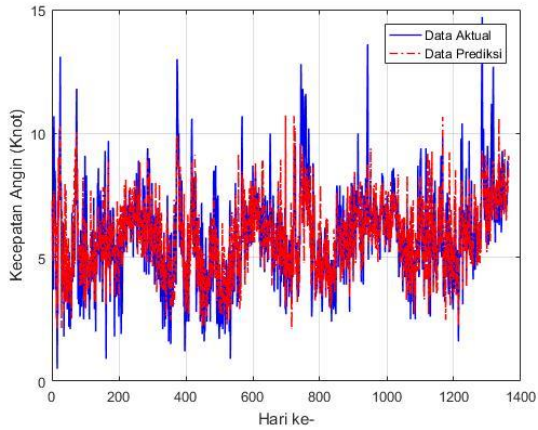


Gambar Lampiran 1. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 1 *hidden node*

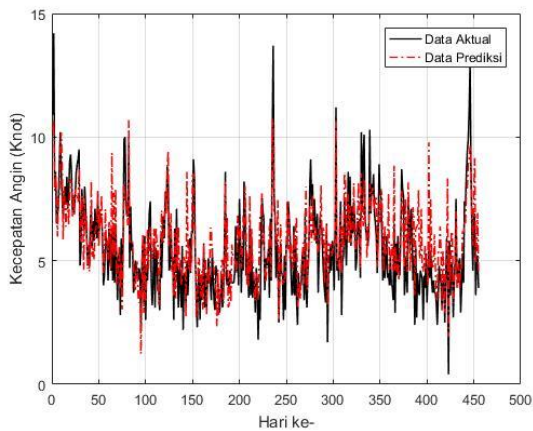


Gambar Lampiran 2. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 1 *hidden node*

- 2 Hidden Node

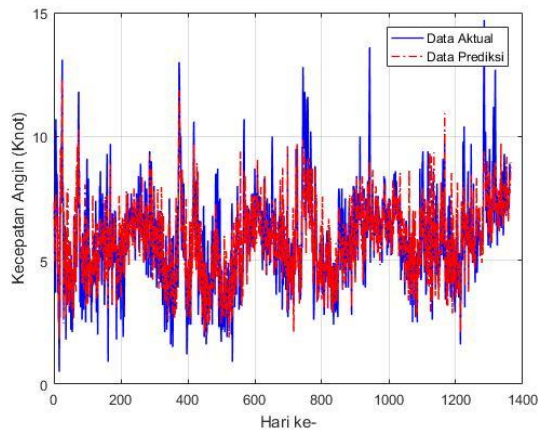


Gambar Lampiran 3. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 2 *hidden node*

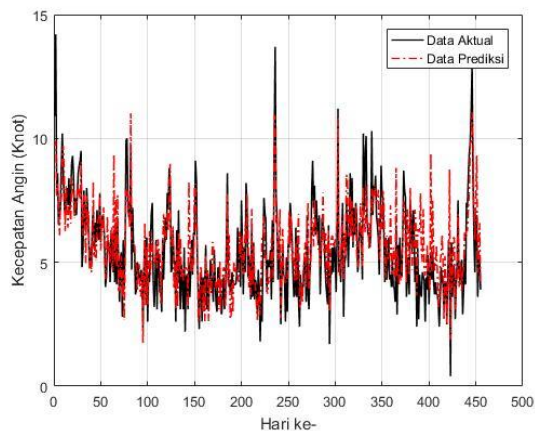


Gambar Lampiran 4. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 2 *hidden node*

- 4 *Hidden Node*

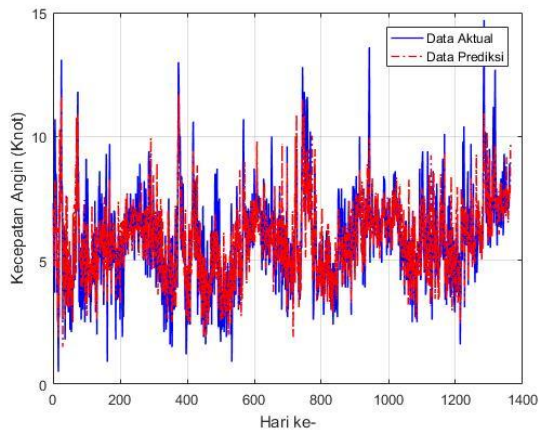


Gambar Lampiran 5. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 4 *hidden node*

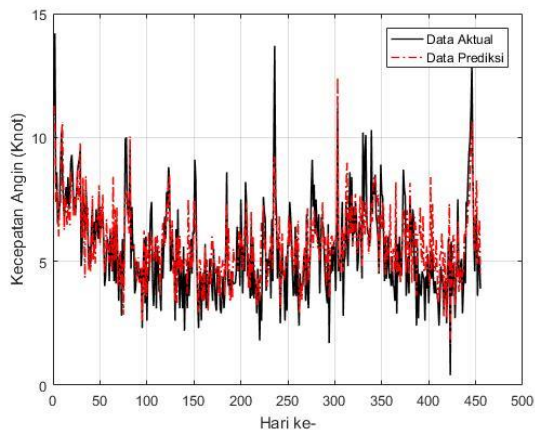


Gambar Lampiran 6. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 4 *hidden node*

- 5 Hidden Node

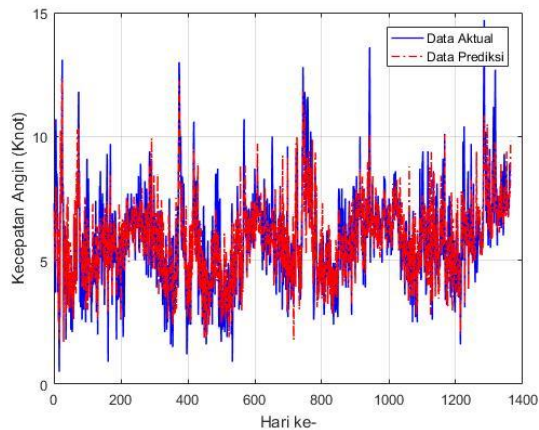


Gambar Lampiran 7. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 5 *hidden node*

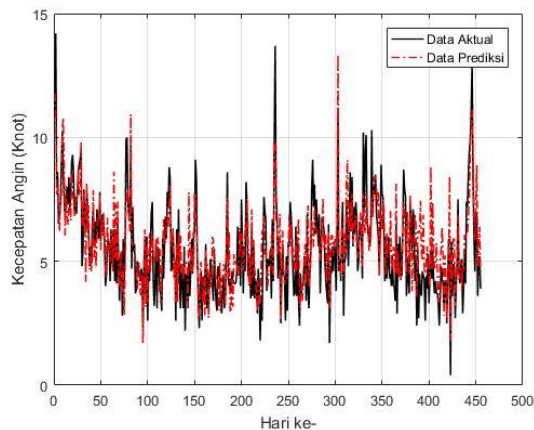


Gambar Lampiran 8. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 5 *hidden node*

- 6 *Hidden Node*

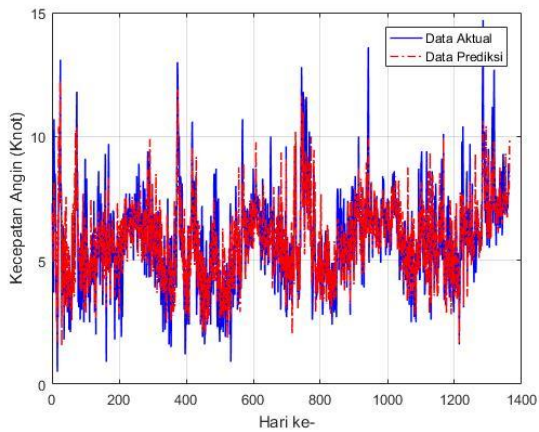


Gambar Lampiran 9. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 6 *hidden node*

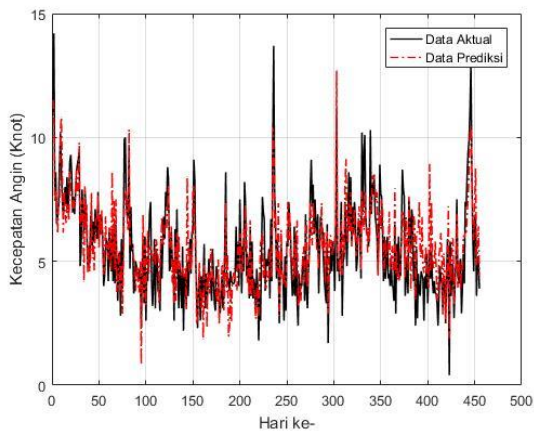


Gambar Lampiran 10. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 6 *hidden node*

- 7 Hidden Node

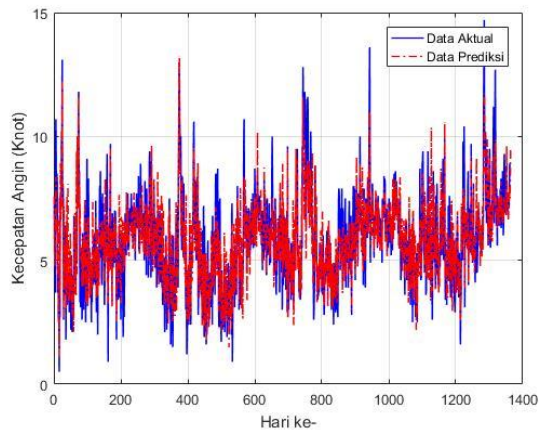


Gambar Lampiran 11. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 7 *hidden node*

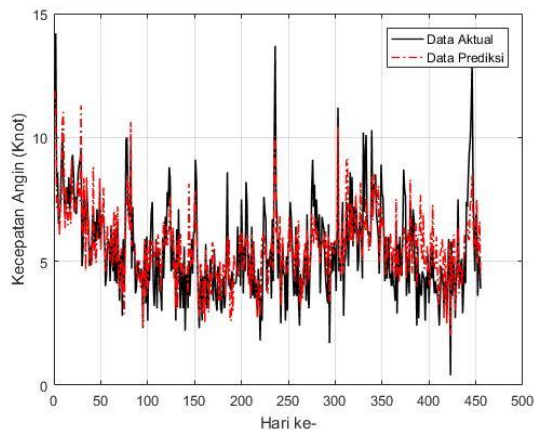


Gambar Lampiran 12. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 7 *hidden node*

- 8 *Hidden Node*

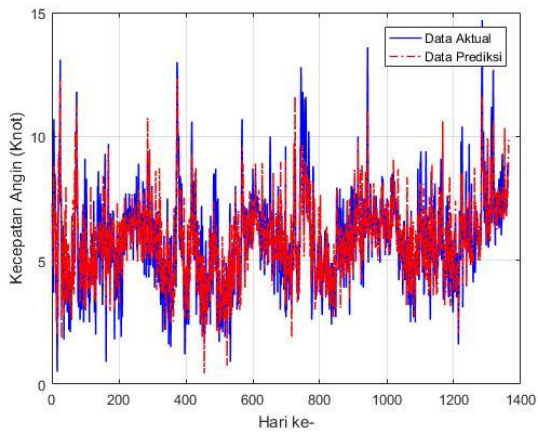


Gambar Lampiran 13. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 8 *hidden node*

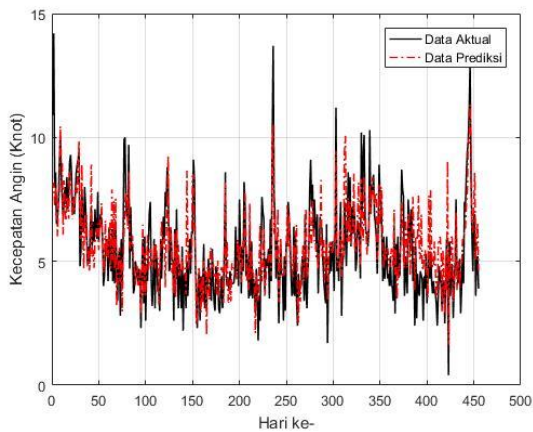


Gambar Lampiran 14. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 8 *hidden node*

- 9 Hidden Node

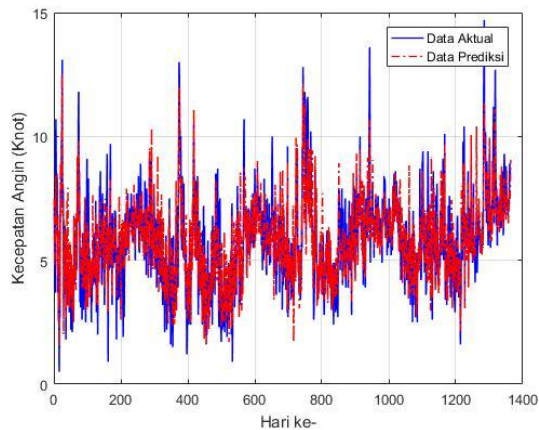


Gambar Lampiran 15. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 9 *hidden node*

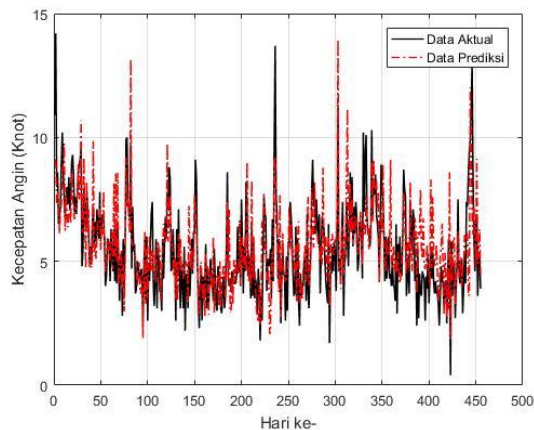


Gambar Lampiran 16. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 9 *hidden node*

- 10 *Hidden Node*



Gambar Lampiran 17. Hasil pelatihan JST untuk data kecepatan angin dengan menggunakan 10 *hidden node*

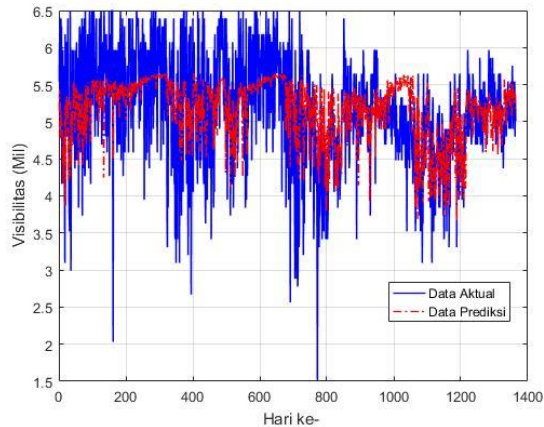


Gambar Lampiran 18. Hasil validasi JST untuk data kecepatan angin dengan menggunakan 10 *hidden node*

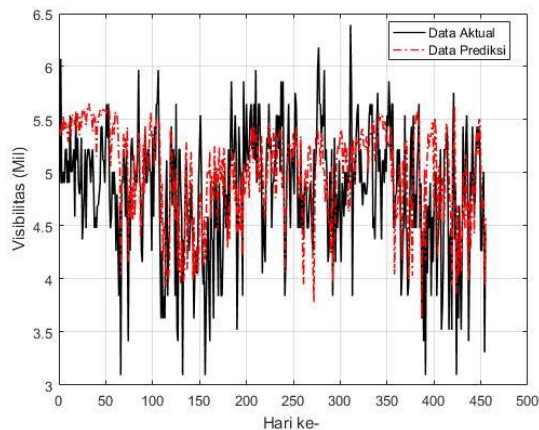
Halaman ini sengaja dikosongkan

Lampiran 4. Grafik Pelatihan dan Validasi Data Prediksi Visibilitas dengan Metode JST

- 1 Hidden Node

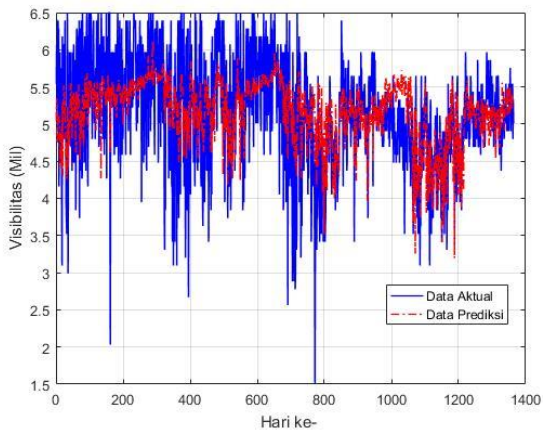


Gambar Lampiran 19. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 1 *hidden node*

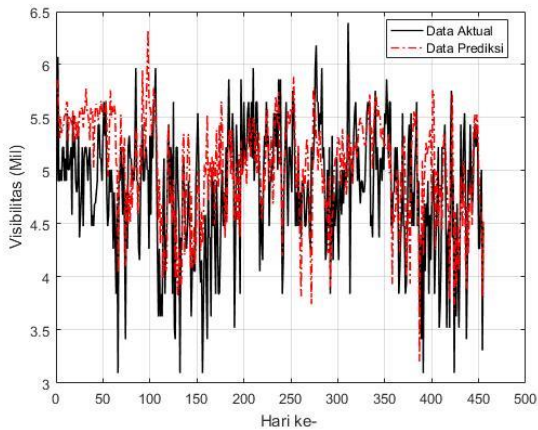


Gambar Lampiran 20. Hasil validasi JST untuk data visibilitas dengan menggunakan 1 *hidden node*

- 2 Hidden Node

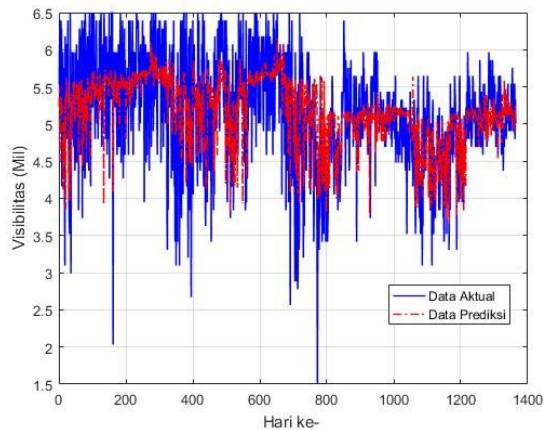


Gambar Lampiran 21. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 2 *hidden node*

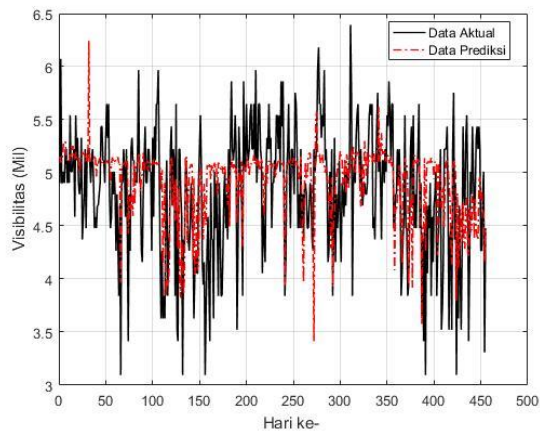


Gambar Lampiran 22. Hasil validasi JST untuk data visibilitas dengan menggunakan 2 *hidden node*

- 3 *Hidden Node*

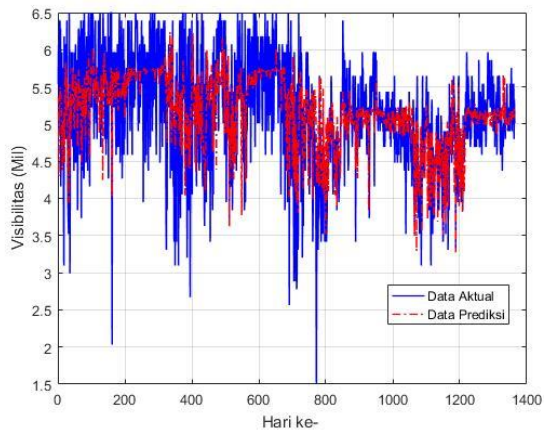


Gambar Lampiran 23. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 3 *hidden node*

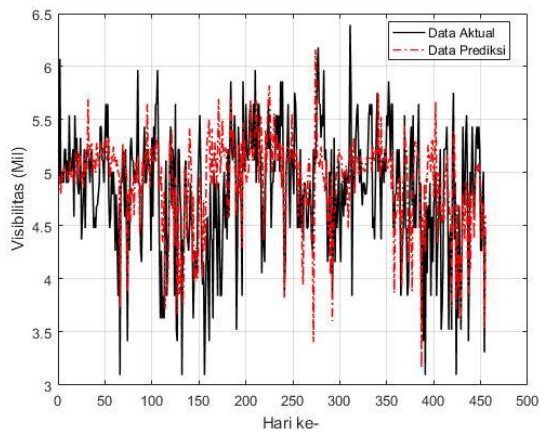


Gambar Lampiran 24. Hasil validasi JST untuk data visibilitas dengan menggunakan 3 *hidden node*

- 5 Hidden Node

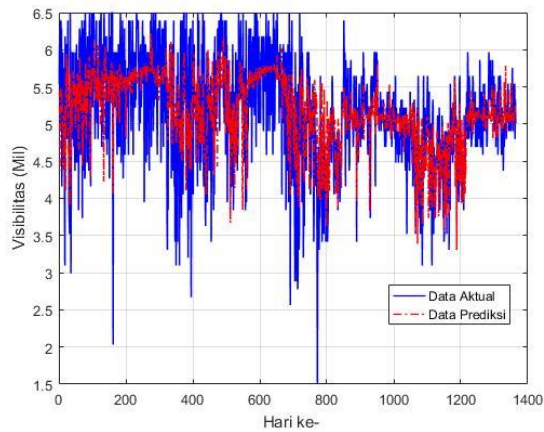


Gambar Lampiran 25. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 5 *hidden node*

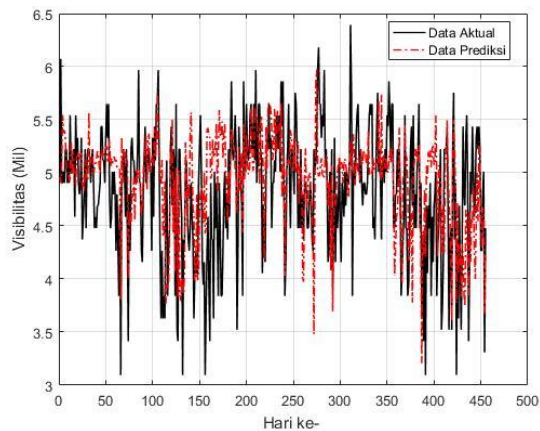


Gambar Lampiran 26. Hasil validasi JST untuk data visibilitas dengan menggunakan 5 *hidden node*

- 6 *Hidden Node*

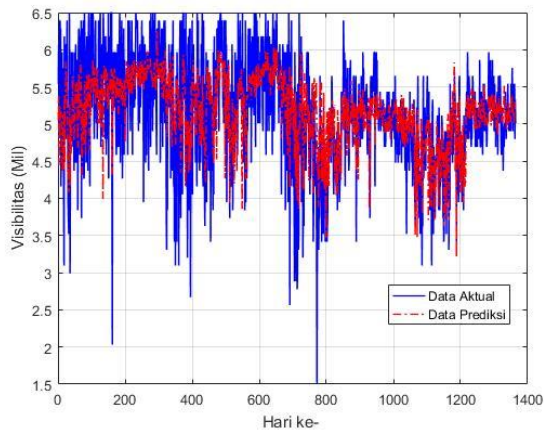


Gambar Lampiran 27. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 6 *hidden node*

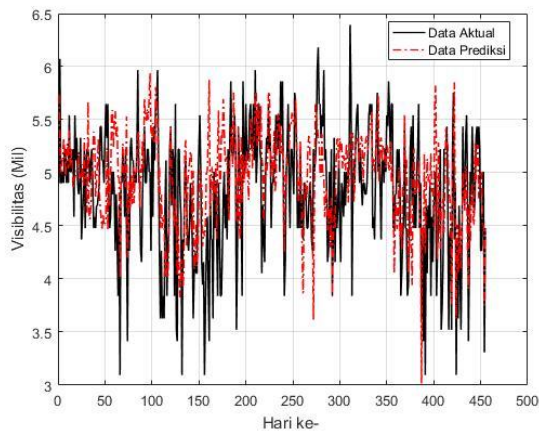


Gambar Lampiran 28. Hasil validasi JST untuk data visibilitas dengan menggunakan 6 *hidden node*

- 7 Hidden Node

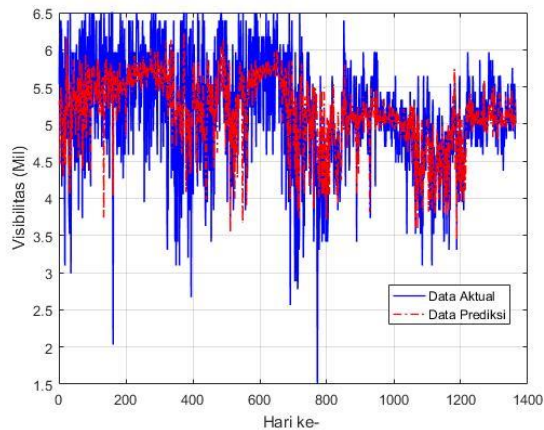


Gambar Lampiran 29. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 7 *hidden node*

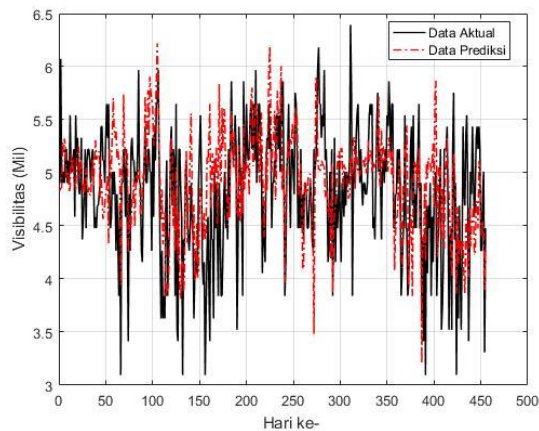


Gambar Lampiran 30. Hasil validasi JST untuk data visibilitas dengan menggunakan 7 *hidden node*

- 8 *Hidden Node*

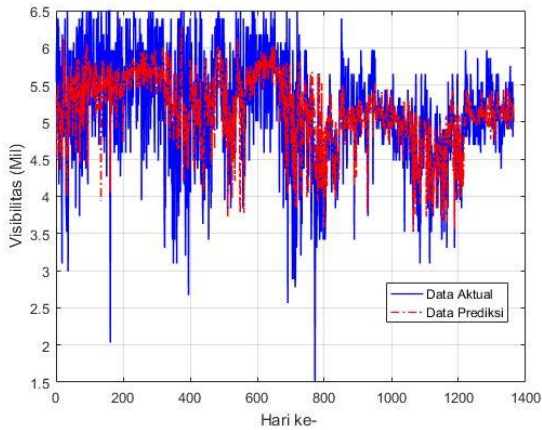


Gambar Lampiran 31. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 8 *hidden node*

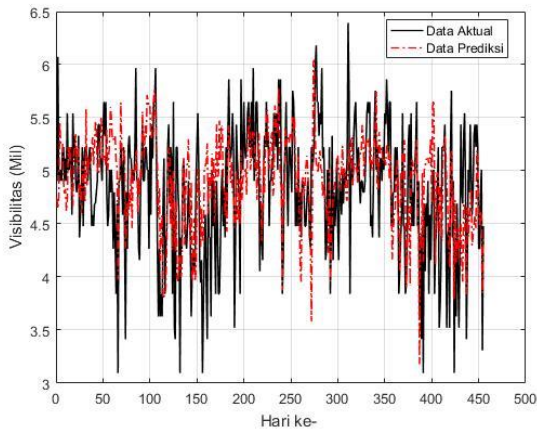


Gambar Lampiran 32. Hasil validasi JST untuk data visibilitas dengan menggunakan 8 *hidden node*

- 9 *Hidden Node*

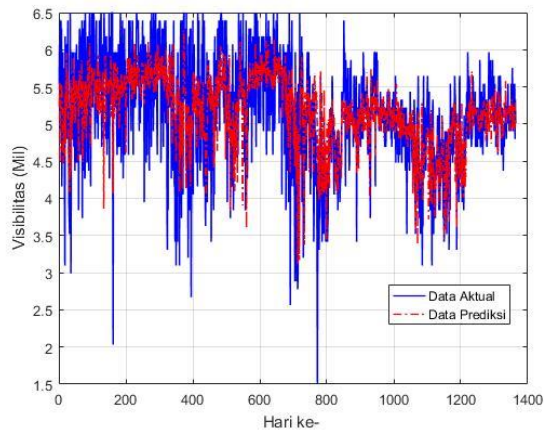


Gambar Lampiran 33. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 9 *hidden node*

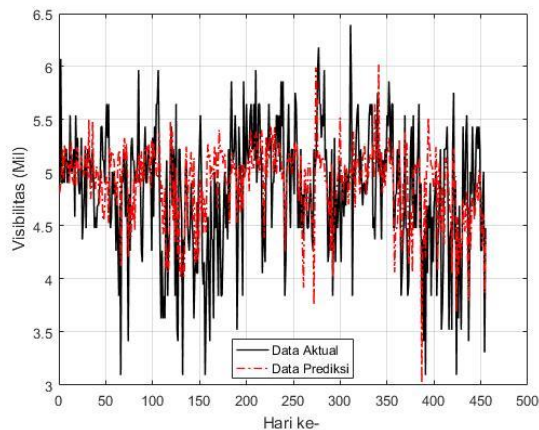


Gambar Lampiran 34. Hasil validasi JST untuk data visibilitas dengan menggunakan 9 *hidden node*

- 10 *Hidden Node*



Gambar Lampiran 35. Hasil pelatihan JST untuk data visibilitas dengan menggunakan 10 *hidden node*



Gambar Lampiran 36. Hasil validasi JST untuk data visibilitas dengan menggunakan 10 *hidden node*

Halaman ini sengaja dikosongkan

Lampiran 5. Kode Program Prediksi Cuaca Metode JST-PSO

```
%PARTICLE SWARM OPTIMIZATION
clc;
clear all;
close all;

% Untuk tiap hidden node
% 1. Ganti fungsi objektif
% 2. Ganti nVar
% 3. Ganti nama untuk save

%% Problem Definition

CostFunction=@(x) (fobpso3(x)); %FUNCTION

nVar=19;           % Number of Decision Variables
VarSize=[1 nVar]; % Size of Decision Variables
Matrix

Min = -1;
Max = 1;
VarMin = repmat(Min,1,nVar); % Lower Bound of
Variables
VarMax = repmat(Max,1,nVar); % Upper Bound of
Variables

%% PSO Parameters

MaxIt=15;          % Maximum Number of Iterations
nPop=50;           % Population Size (Swarm Size)

% PSO Parameters
w=1;               % Inertia Weight
wdamp=0.99;        % Inertia Weight Damping Ratio
c1=1.5;            % Personal Learning Coefficient
c2=2.0;            % Global Learning Coefficient
% Velocity Limits
VelMax=0.1*(VarMax-VarMin);
```

```

VelMin=-VelMax;

%% Initialization

empty_particle.Position=[];
empty_particle.Cost=[];
empty_particle.Velocity=[];
empty_particle.Best.Position=[];
empty_particle.Best.Cost=[];

particle= repmat(empty_particle,nPop,1);

GlobalBest.Cost=inf;

for i=1:nPop

    % Initialize Position

    particle(i).Position=unifrnd(VarMin,VarMax,VarSize);

    % Initialize Velocity
    particle(i).Velocity=zeros(VarSize);

    % Evaluation

    particle(i).Cost=CostFunction(particle(i).Position);

    % Update Personal Best

    particle(i).Best.Position=particle(i).Position;
    particle(i).Best.Cost=particle(i).Cost;

    % Update Global Best
    if particle(i).Best.Cost<GlobalBest.Cost

        GlobalBest=particle(i).Best;
    end
end

```



```

        end

    end

    BestCost=zeros(MaxIt,1);

    %% PSO Main Loop

    for it=1:MaxIt

        for i=1:nPop

            % Update Velocity
            particle(i).Velocity =
w*particle(i).Velocity ...

+c1*rand(VarSize).*(particle(i).Best.Position-
particle(i).Position) ...

+c2*rand(VarSize).*(GlobalBest.Position-
particle(i).Position);

            % Apply Velocity Limits
            particle(i).Velocity =
max(particle(i).Velocity,VelMin);
            particle(i).Velocity =
min(particle(i).Velocity,VelMax);

            % Update Position
            particle(i).Position =
particle(i).Position + particle(i).Velocity;

            % Velocity Mirror Effect
            IsOutside=(particle(i).Position<VarMin |
particle(i).Position>VarMax);
            particle(i).Velocity(IsOutside)=-
particle(i).Velocity(IsOutside);

```

```

        % Apply Position Limits
        particle(i).Position =
max(particle(i).Position,VarMin);
        particle(i).Position =
min(particle(i).Position,VarMax);

        % Evaluation
        particle(i).Cost =
CostFunction(particle(i).Position);

        % Update Personal Best
        if
particle(i).Cost<particle(i).Best.Cost

particle(i).Best.Position=particle(i).Position;

particle(i).Best.Cost=particle(i).Cost;

        % Update Global Best
        if
particle(i).Best.Cost<GlobalBest.Cost

                GlobalBest=particle(i).Best;

        end

    end

end

BestCost(it)=GlobalBest.Cost;

disp(['Iteration ' num2str(it) ' : Best Cost
= ' num2str(BestCost(it))]);

w=w*wdamp;

```

```

end

RMSE_Value = GlobalBest

%% Results

figure;
plot(BestCost, 'LineWidth', 2);
% semilogy(BestCost, 'LineWidth', 2);
xlabel('Iteration');
ylabel('Best Cost');
grid on;

save ('psol_hn10.mat') % SAVE ALL -----

```

Halaman ini sengaja dikosongkan

Lampiran 6. Kode Program Fungsi Objektif untuk PSO

```
function RMSE=fobpsol0(x)
% variabelinput = xlsread('D:\Undergraduate
Thesis\Data Cuaca\Data\Data Juanda
2.1_Visibilitas.xlsx','Visibility','H2:K1366');
% validasi = xlsread('D:\Undergraduate
Thesis\Data Cuaca\Data\Data Juanda
2.1_Visibilitas.xlsx','Visibility','L2:L1366');

variabelinput = xlsread('D:\Undergraduate
Thesis\Data Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx','Wind','H2:K1366');
validasi = xlsread('D:\Undergraduate Thesis\Data
Cuaca\Data\Data Juanda 2.1_Kecepatan
Angin.xlsx','Wind','L2:L1366');

data = 1365; % Jumlah Data

% 1 individu = seluruh bobot
t1=x(1);
t2=x(2);
t3=x(3);
t4=x(4);
t5=x(5);

t6=x(6);
t7=x(7);
t8=x(8);
t9=x(9);
t10=x(10);

t11=x(11);
t12=x(12);
t13=x(13);
t14=x(14);
t15=x(15);

t16=x(16);
```

```
t17=x (17) ;  
t18=x (18) ;  
t19=x (19) ;  
t20=x (20) ;
```

```
t21=x (21) ;  
t22=x (22) ;  
t23=x (23) ;  
t24=x (24) ;  
t25=x (25) ;
```

```
t26=x (26) ;  
t27=x (27) ;  
t28=x (28) ;  
t29=x (29) ;  
t30=x (30) ;
```

```
t31=x (31) ;  
t32=x (32) ;  
t33=x (33) ;  
t34=x (34) ;  
t35=x (35) ;
```

```
t36=x (36) ;  
t37=x (37) ;  
t38=x (38) ;  
t39=x (39) ;  
t40=x (40) ;
```

```
t41=x (41) ;  
t42=x (42) ;  
t43=x (43) ;  
t44=x (44) ;  
t45=x (45) ;
```

```
t46=x (46) ;  
t47=x (47) ;  
t48=x (48) ;  
t49=x (49) ;
```

```

t50=x(50);

t51=x(51);
t52=x(52);
t53=x(53);
t54=x(54);
t55=x(55);
t56=x(56);
t57=x(57);
t58=x(58);
t59=x(59);
t60=x(60);
t61=x(61);

%Input ke hidden
a1 = variabelinput(1:data,1)*t1;
b1 = variabelinput(1:data,2)*t2;
c1 = variabelinput(1:data,3)*t3;
d1 = variabelinput(1:data,4)*t4;
e1 = ones(data,1)*t5;

hn1 = [a1 b1 c1 d1 e1];

a2 = variabelinput(1:data,1)*t6;
b2 = variabelinput(1:data,2)*t7;
c2 = variabelinput(1:data,3)*t8;
d2 = variabelinput(1:data,4)*t9;
e2 = ones(data,1)*t10;

hn2 = [a2 b2 c2 d2 e2];

a3 = variabelinput(1:data,1)*t11;
b3 = variabelinput(1:data,2)*t12;
c3 = variabelinput(1:data,3)*t13;
d3 = variabelinput(1:data,4)*t14;
e3 = ones(data,1)*t15;

hn3 = [a3 b3 c3 d3 e3];

```

```
a4 = variabelinput(1:data,1)*t16;  
b4 = variabelinput(1:data,2)*t17;  
c4 = variabelinput(1:data,3)*t18;  
d4 = variabelinput(1:data,4)*t19;  
e4 = ones(data,1)*t20;
```

```
hn4 = [a4 b4 c4 d4 e4];
```

```
a5 = variabelinput(1:data,1)*t21;  
b5 = variabelinput(1:data,2)*t22;  
c5 = variabelinput(1:data,3)*t23;  
d5 = variabelinput(1:data,4)*t24;  
e5 = ones(data,1)*t25;
```

```
hn5 = [a5 b5 c5 d5 e5];
```

```
a6 = variabelinput(1:data,1)*t26;  
b6 = variabelinput(1:data,2)*t27;  
c6 = variabelinput(1:data,3)*t28;  
d6 = variabelinput(1:data,4)*t29;  
e6 = ones(data,1)*t30;
```

```
hn6 = [a6 b6 c6 d6 e6];
```

```
a7 = variabelinput(1:data,1)*t31;  
b7 = variabelinput(1:data,2)*t32;  
c7 = variabelinput(1:data,3)*t33;  
d7 = variabelinput(1:data,4)*t34;  
e7 = ones(data,1)*t35;
```

```
hn7 = [a7 b7 c7 d7 e7];
```

```
a8 = variabelinput(1:data,1)*t36;  
b8 = variabelinput(1:data,2)*t37;  
c8 = variabelinput(1:data,3)*t38;  
d8 = variabelinput(1:data,4)*t39;  
e8 = ones(data,1)*t40;
```



```

hn8 = [a8 b8 c8 d8 e8];

a9 = variabelinput(1:data,1)*t41;
b9 = variabelinput(1:data,2)*t42;
c9 = variabelinput(1:data,3)*t43;
d9 = variabelinput(1:data,4)*t44;
e9 = ones(data,1)*t45;

hn9 = [a9 b9 c9 d9 e9];

a10 = variabelinput(1:data,1)*t46;
b10 = variabelinput(1:data,2)*t47;
c10 = variabelinput(1:data,3)*t48;
d10 = variabelinput(1:data,4)*t49;
e10 = ones(data,1)*t50;

hn10 = [a10 b10 c10 d10 e10];

% Fungsi aktivasi hidden node
for i = 1:data;
    nilaihn1(i,1) = sum(hn1(i,1:5));
end;

for i = 1:data;
    nilaihn2(i,1) = sum(hn2(i,1:5));
end;

for i = 1:data;
    nilaihn3(i,1) = sum(hn3(i,1:5));
end;

for i = 1:data;
    nilaihn4(i,1) = sum(hn4(i,1:5));
end;

for i = 1:data;
    nilaihn5(i,1) = sum(hn5(i,1:5));
end;

```

```

for i = 1:data;
    nilaihn6(i,1) = sum(hn6(i,1:5));
end;

for i = 1:data;
    nilaihn7(i,1) = sum(hn7(i,1:5));
end;

for i = 1:data;
    nilaihn8(i,1) = sum(hn8(i,1:5));
end;

for i = 1:data;
    nilaihn9(i,1) = sum(hn9(i,1:5));
end;

for i = 1:data;
    nilaihn10(i,1) = sum(hn10(i,1:5));
end;

% Hidden ke output
fhn1 = tansig(nilaihn1)*t51;
fhn2 = tansig(nilaihn2)*t52;
fhn3 = tansig(nilaihn3)*t53;
fhn4 = tansig(nilaihn4)*t54;
fhn5 = tansig(nilaihn5)*t55;
fhn6 = tansig(nilaihn6)*t56;
fhn7 = tansig(nilaihn7)*t57;
fhn8 = tansig(nilaihn8)*t58;
fhn9 = tansig(nilaihn9)*t59;
fhn10 = tansig(nilaihn10)*t60;
fhn11 = ones(data,1)*t61; %bias pada
output layer

matrix
=[fhn1,fhn2,fhn3,fhn4,fhn5,fhn6,fhn7,fhn8,fhn9,f
hn10,fhn11];

```

```

    for i = 1:data;
        nilaion(i,1)= sum(matrix(i,1:11));
    end;

% Fungsi aktivasi output layer

    prediksi = tansig(nilaion);

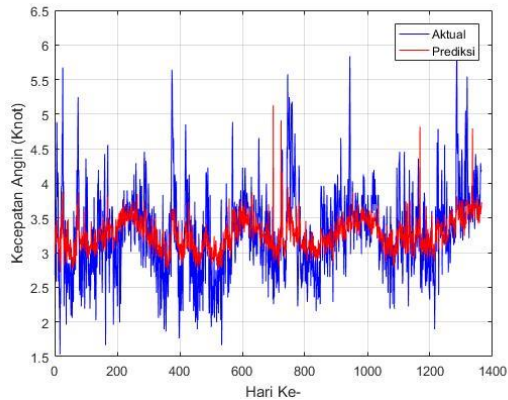
% Objective function
    selisih = prediksi - validasi;
    selisihkuadrat = selisih.^2;
    RMSE = sqrt((sum(selisihkuadrat))/data);

```

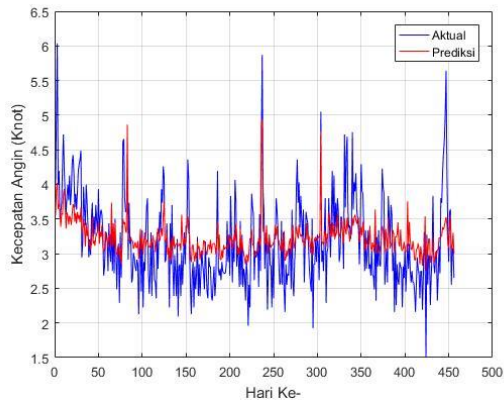
Halaman ini sengaja dikosongkan

Lampiran 7. Grafik Pelatihan dan Validasi Data Prediksi Kecepatan Angin dengan Metode Gabungan JST-PSO

- 1 *Hidden Node*

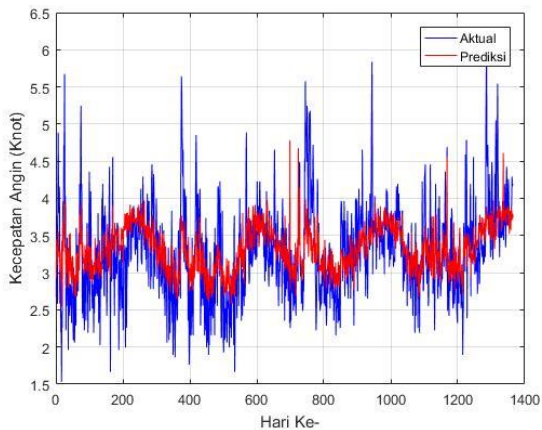


Gambar Lampiran 37. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 1 *hidden node*

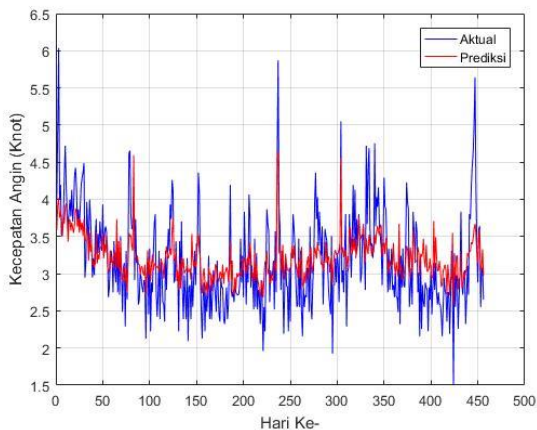


Gambar Lampiran 38. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 1 *hidden node*

- 2 Hidden Node

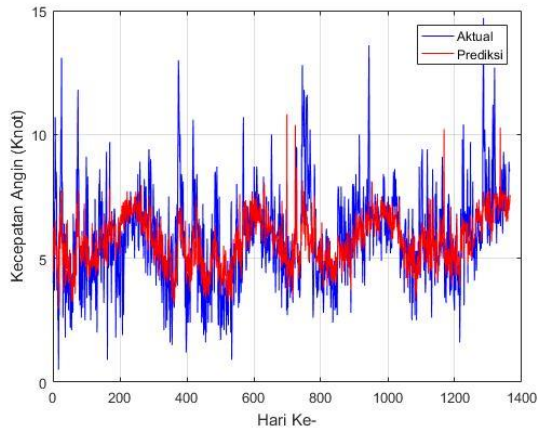


Gambar Lampiran 39. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 2 *hidden node*

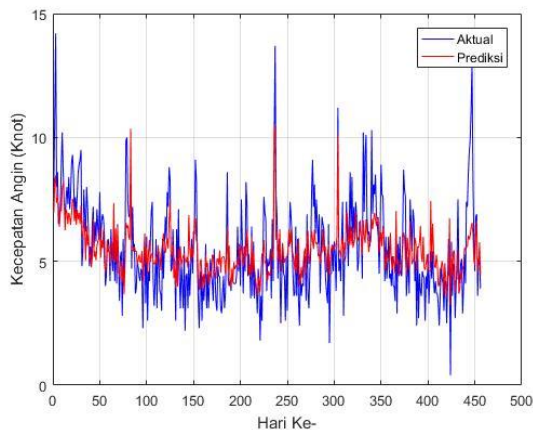


Gambar Lampiran 40. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 2 *hidden node*

- 3 *Hidden Node*

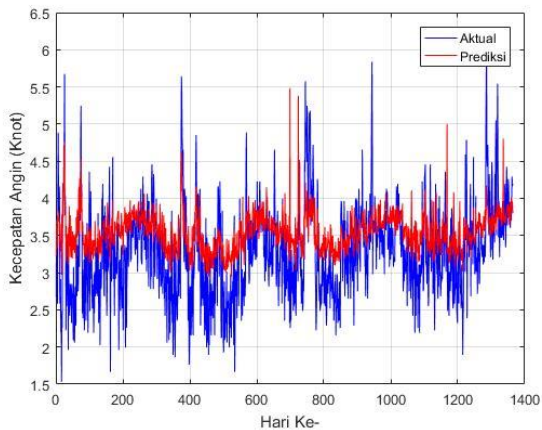


Gambar Lampiran 41. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 3 *hidden node*

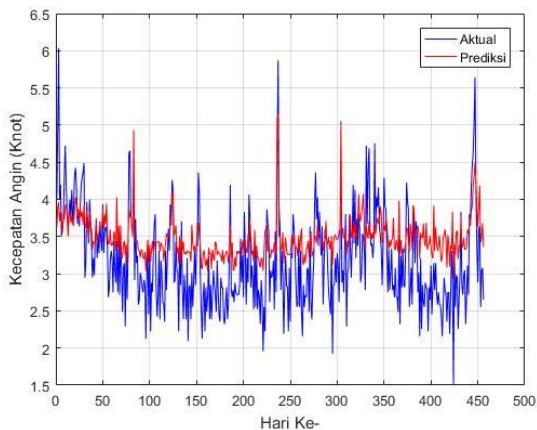


Gambar Lampiran 42. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 3 *hidden node*

- 4 *Hidden Node*

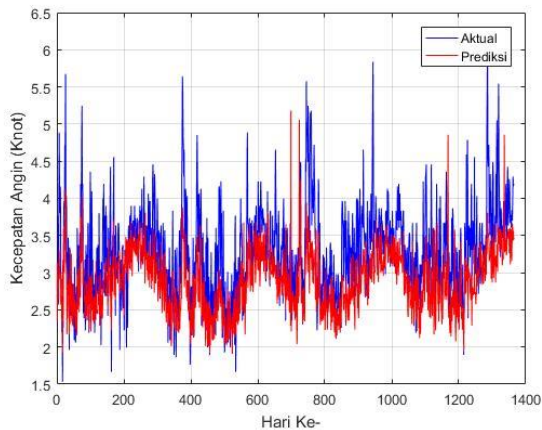


Gambar Lampiran 43. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 4 *hidden node*

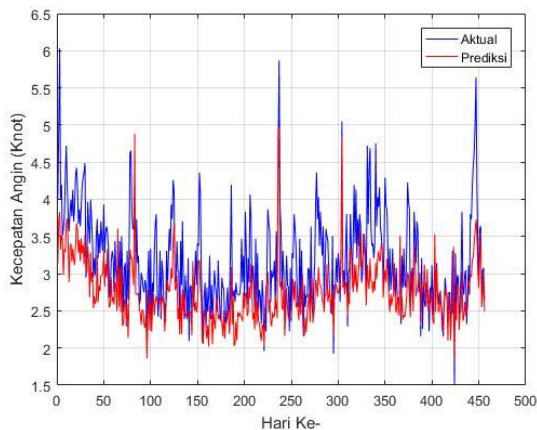


Gambar Lampiran 44. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 4 *hidden node*

- 5 Hidden Node

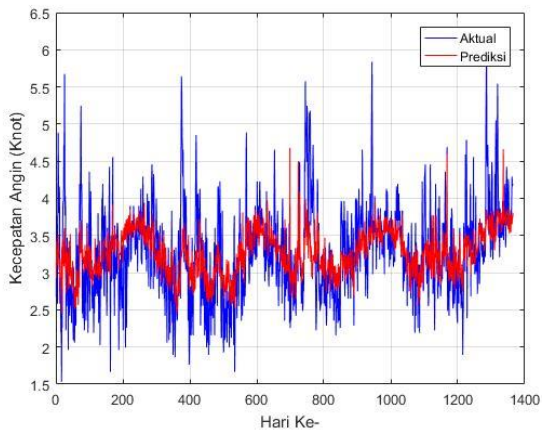


Gambar Lampiran 45. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 5 *hidden node*

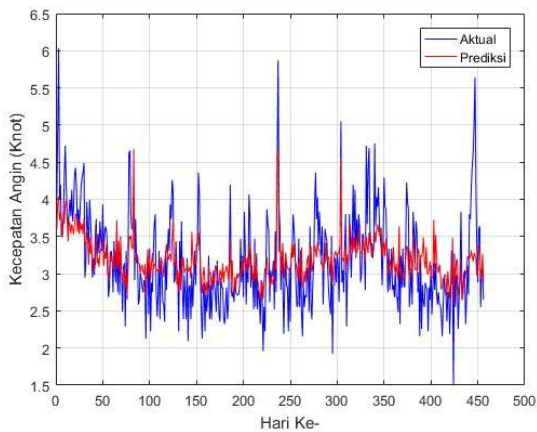


Gambar Lampiran 46. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 5 *hidden node*

- 7 Hidden Node

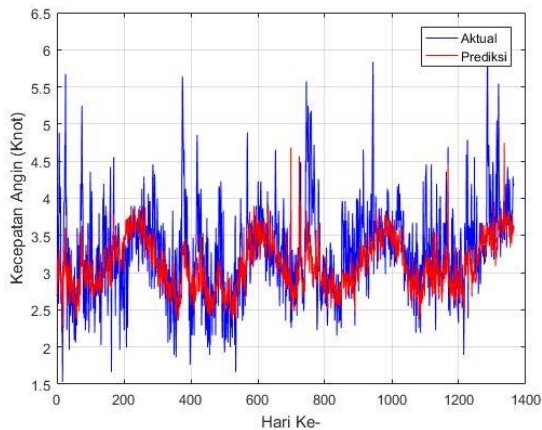


Gambar Lampiran 47. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 7 *hidden node*

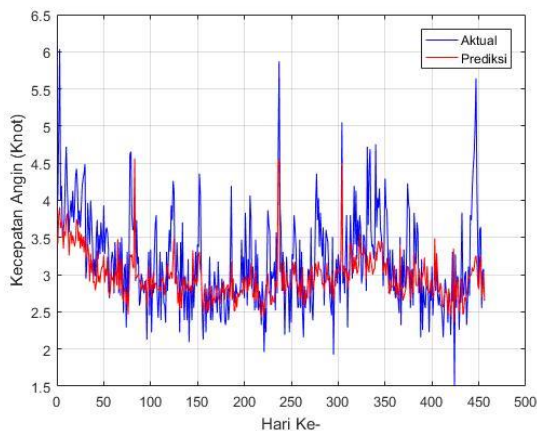


Gambar Lampiran 48. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 7 *hidden node*

- 8 *Hidden Node*

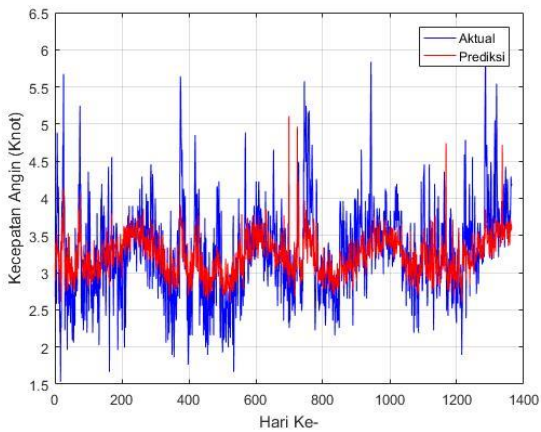


Gambar Lampiran 49. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 8 *hidden node*

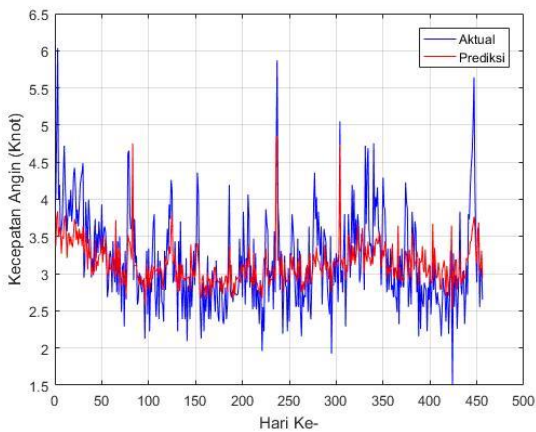


Gambar Lampiran 50. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 8 *hidden node*

- 9 *Hidden Node*

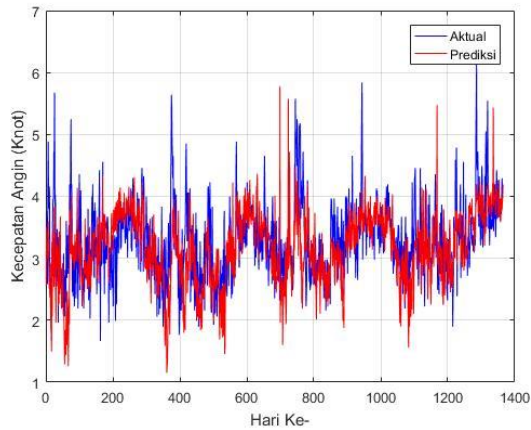


Gambar Lampiran 51. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 9 *hidden node*

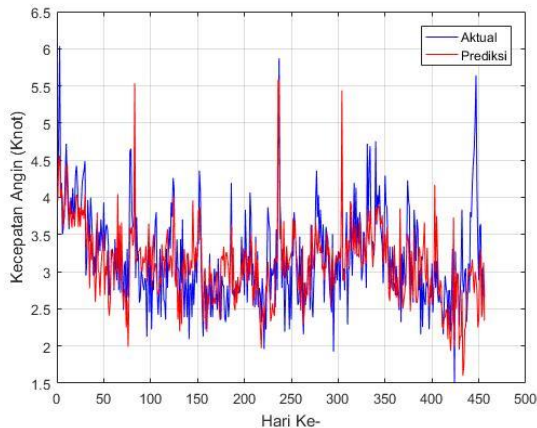


Gambar Lampiran 52. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 9 *hidden node*

- 10 *Hidden Node*



Gambar Lampiran 53. Hasil pelatihan JST-PSO untuk data kecepatan angin dengan menggunakan 10 *hidden node*

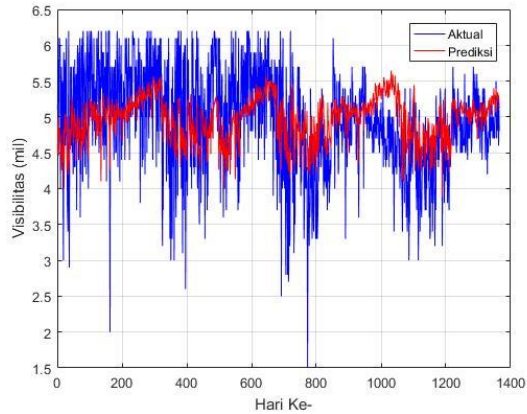


Gambar Lampiran 54. Hasil validasi JST-PSO untuk data kecepatan angin dengan menggunakan 10 *hidden node*

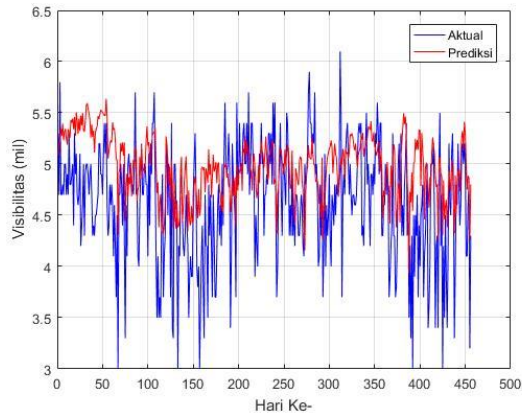
Halaman ini sengaja dikosongkan

Lampiran 8. Grafik Pelatihan dan Validasi Data Prediksi Visibilitas dengan Metode Gabungan JST-PSO

- 1 *Hidden Node*

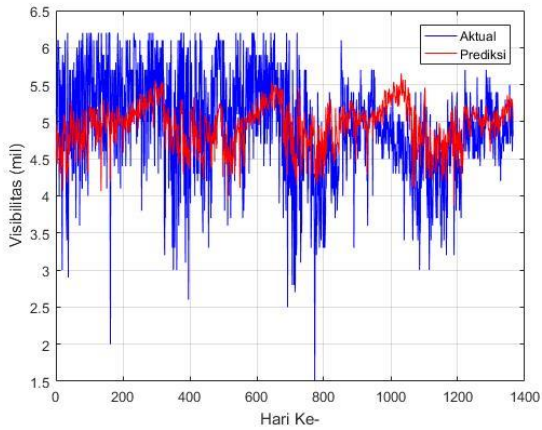


Gambar Lampiran 55. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 1 *hidden node*

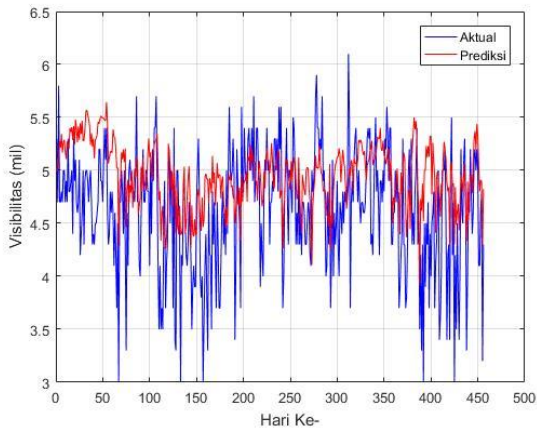


Gambar Lampiran 56. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 1 *hidden node*

- 2 Hidden Node

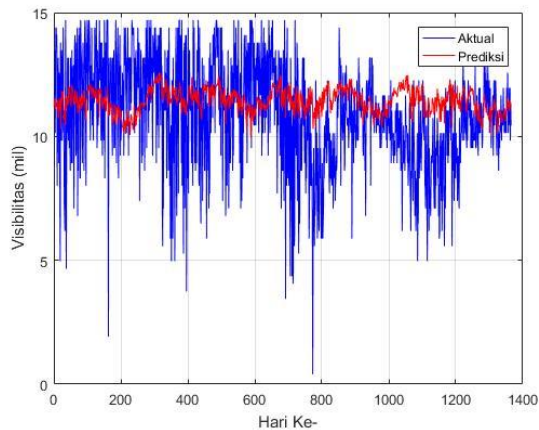


Gambar Lampiran 57. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 2 *hidden node*

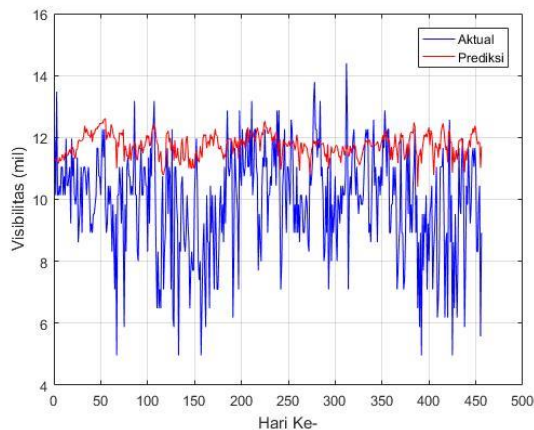


Gambar Lampiran 58. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 2 *hidden node*

- 3 *Hidden Node*

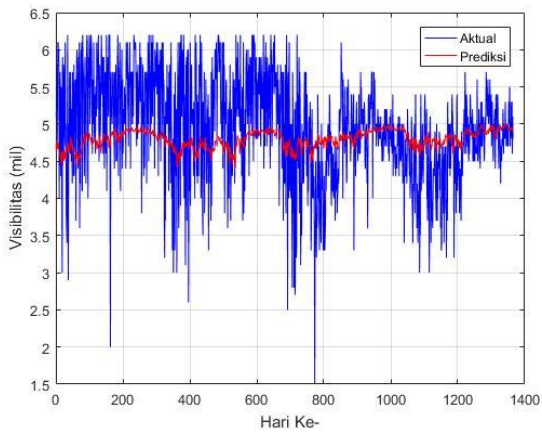


Gambar Lampiran 59. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 3 *hidden node*

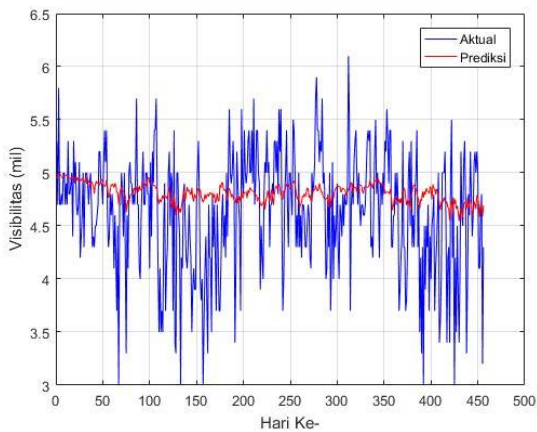


Gambar Lampiran 60. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 3 *hidden node*

- 4 Hidden Node

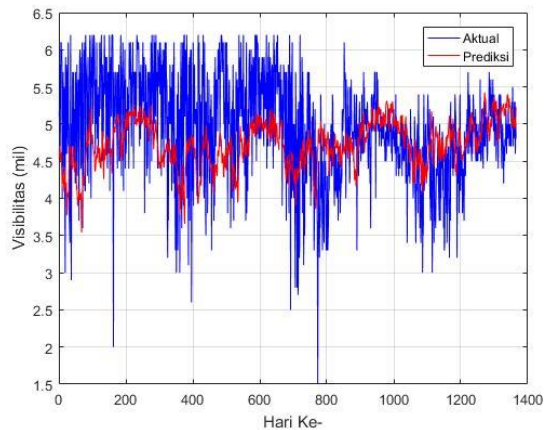


Gambar Lampiran 61. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 4 *hidden node*

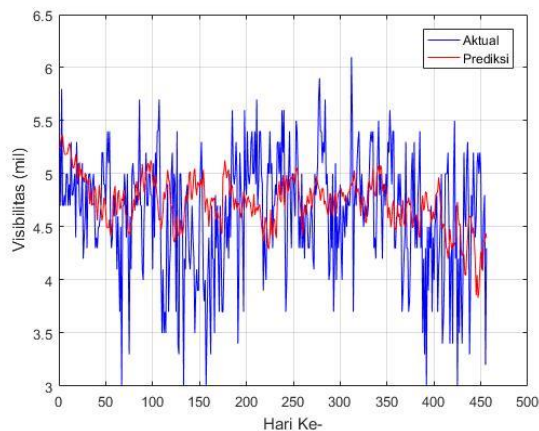


Gambar Lampiran 62. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 4 *hidden node*

- 5 Hidden Node

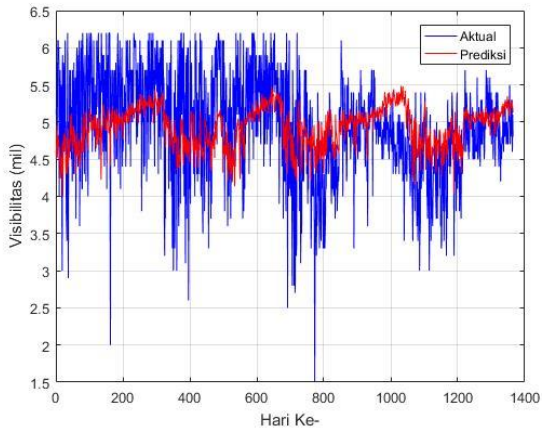


Gambar Lampiran 63. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 5 *hidden node*

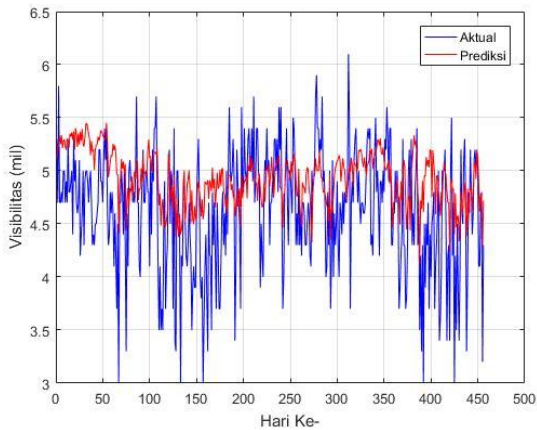


Gambar Lampiran 64. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 5 *hidden node*

- 6 Hidden Node

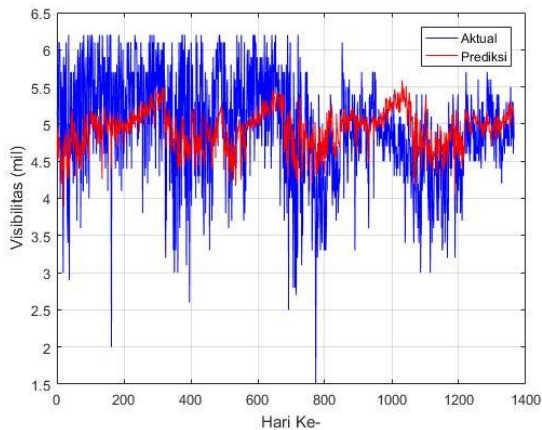


Gambar Lampiran 65. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 6 *hidden node*

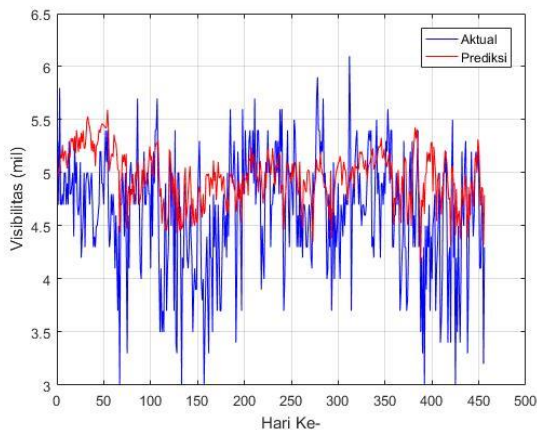


Gambar Lampiran 66. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 6 *hidden node*

- 7 Hidden Node

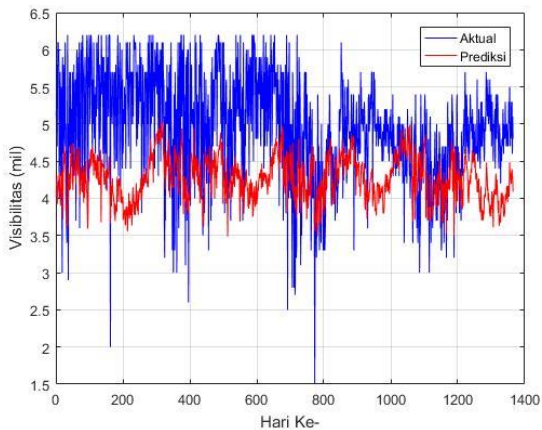


Gambar Lampiran 67. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 7 *hidden node*

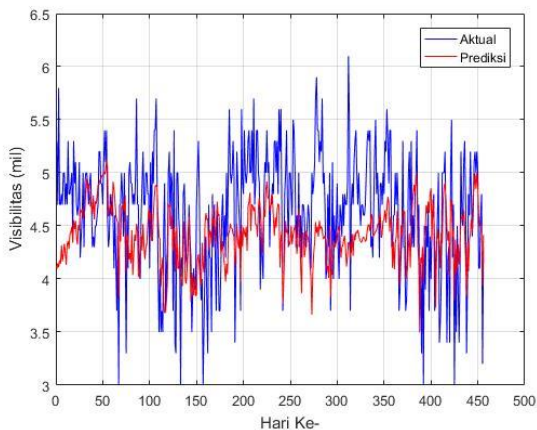


Gambar Lampiran 68. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 7 *hidden node*

- 8 *Hidden Node*

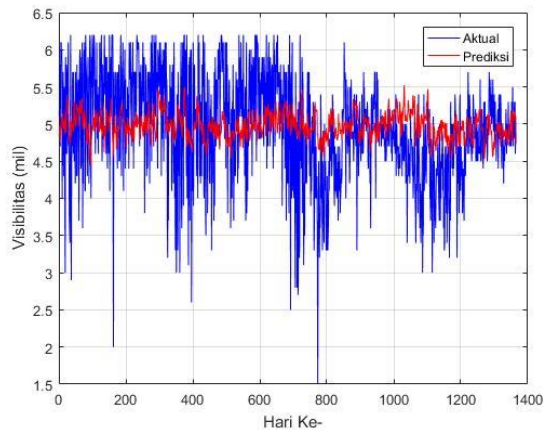


Gambar Lampiran 69. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 8 *hidden node*

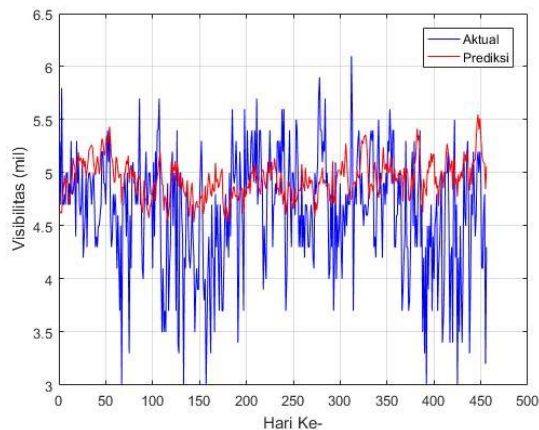


Gambar Lampiran 70. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 8 *hidden node*

- 10 *Hidden Node*



Gambar Lampiran 71. Hasil pelatihan JST-PSO untuk data visibilitas dengan menggunakan 10 *hidden node*



Gambar Lampiran 72. Hasil validasi JST-PSO untuk data visibilitas dengan menggunakan 10 *hidden node*

Halaman ini sengaja dikosongkan

Lampiran 9. Data Cuaca yang Digunakan Untuk Melakukan Prediksi Cuaca

Data cuaca pada penelitian tugas akhir ini diambil dari *National Climatic Data Center* pada tanggal 27 April 2017 pukul 22.10 WIB. Berikut ini langkah-langkah pengambilan data dalam penelitian tugas akhir ini.

1. Masuk ke alamat www7.ncdc.noaa.gov/CDO/dataproduct

2. Kemudian pilih “Surface Data, Global Summary of the Day”, lalu klik “Access Data/Products”.

3. Pilih negara dimana stasiun cuaca yang dituju berada pada baris “Country”. Kemudian pilih negara “Indonesia”, dan klik continue.

Global Summary of the Day (GSOD)

Retrieve data for:

☒ **Selected INDONESIA stations** - Note: may be slow to load station list on next page

4. Muncul informasi mengenai stasiun cuaca tersebut. Klik continue.

Global Summary of the Day (GSOD)

Selected INDONESIA stations - Note: may be slow to load station list on next page

Select Stations:

SULTAN ISKANDARMUDA.....	96011099999	01/1973	to	06/2017	▲
SULTAN MAHMUD BADARUDDIN II.....	96221099999	05/1962	to	06/2017	
SULTAN SYARIF KASIM II.....	96109099999	01/1959	to	06/2017	
SULTAN THAHA.....	96195099999	01/1973	to	06/2017	
SUMBANA BESAR.....	97260099999	02/1964	to	06/2017	
SUPADIO.....	96581099999	01/1973	to	06/2017	
SURABAYA JUANDA.....	96935099999	01/1973	to	06/2017	
SURABAYA/GEDANGAN.....	96937099999	10/1980	to	06/2017	
SURABAYA/PERAK.....	96933099999	05/1949	to	06/2017	
SUSILO.....	96559099999	12/1979	to	06/2017	
SYAMSUDIN NOOR.....	96685099999	10/1962	to	06/2017	▼

[Order by Station ID](#) - [Order by Station Name](#)

5. Pilih “Surabaya Juanda” jika ingin mengambil data harian pada stasiun cuaca pada Bandara Internasional Juanda. Kemudian klik continue.

Global Summary of the Day (GSOD)

Select Date Restrictions:

☒ **Use Date Range**

<== OR ==>

☐ **Use Selected Dates**

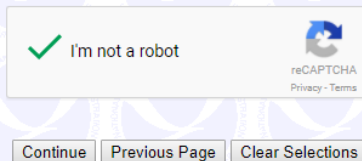
	Year	Month	Day
From	2016 ▼	11 ▼	01 ▼
To	2016 ▼	12 ▼	31 ▼

Year	Month	Day	Hour
1973 ▲	01 ▲	01 ▲	
1974 ▲	02 ▲	02 ▲	
1975 ▲	03 ▲	03 ▲	
1976 ▲	04 ▲	04 ▲	
1977 ▲	05 ▲	05 ▲	
1978 ▲	06 ▲	06 ▲	
1979 ▲	07 ▲	07 ▲	
1980 ▲	08 ▲	08 ▲	
1981 ▲	09 ▲	09 ▲	
1982 ▼	10 ▼	10 ▼	

Tabular Data Output

Select Output Format:

Space Delimited ▼



6. Kemudian pilih tanggal untuk data harian. Jika ingin mengambil banyak data dalam jangka waktu tertentu, pilih “Use Date Range”. Kemudian klik continue.

NNDC CLIMATE DATA ONLINE

Climate Data Online, Data Output

[CDO3714277382209.txt](#)

7. Muncul tautan yang dapat diunduh dalam format .txt. Klik tautan tersebut dan akan terunduh file yang berisi data cuaca yang diinginkan.

Berikut ini merupakan kutipan data cuaca harian untuk tanggal 1 - 31 Desember 2016, yang diambil dari *National Climatic Data Center*.

Tabel 13. Tabel Cuaca Harian Tanggal 1 - 31 Desember 2016

TANGGAL	TEMP (°F)	DEWP (°F)	SLP (Milibar)	STP (Milibar)	VISIB (Mil)	MXSPD (Knot)	WDSP (Knot)
01-12-16	82.7	75.7	1009.2	1008.9	7	4.2	3.6
02-12-16	79.9	75.2	1008.9	1008.6	11.1	3.5	4.9
03-12-16	80.5	75	1008.9	1008.6	11.1	4.5	4.8
04-12-16	81.4	76	1007.9	1007.5	6	3.9	2.7
05-12-16	79.7	75.9	1007.6	1007.2	8	3.4	4
06-12-16	81.1	75.6	1007.5	1007.2	8	4.1	5
07-12-16	83	75.6	1007.4	1007	13	5.2	7.5
08-12-16	82.6	76.2	1006.5	1006.2	7	4.6	4.5
09-12-16	82.5	76.9	1006.6	1006.3	8	4.5	4.1
10-12-16	83.6	75.3	1007.4	1007	8.9	5.1	4.9
11-12-16	82.5	74.8	1008.2	1007.8	7	5.3	5.1
12-12-16	81.1	75.6	1009.3	1009	8	4.7	2.9
13-12-16	78.7	74.6	1009.5	1009.1	8	3.3	4
14-12-16	82.2	74.5	1009.1	1008.8	11.1	4	5.3
15-12-16	80.3	75.4	1009.2	1008.8	11.1	4.8	4.1
16-12-16	81.9	74.7	1008.3	1008	12	4.6	7.4
17-12-16	82.6	74.9	1008.7	1008.3	12	5.2	7.2
18-12-16	83.4	75.8	1008	1007.7	15	4.8	8.5
19-12-16	81.7	75.7	1007.1	1006.8	15	4.4	9.4
20-12-16	83.6	75.8	1005.6	1005.2	17.1	4.6	9.9
21-12-16	84.2	73.4	1006	1005.7	17.1	5.1	11.1
22-12-16	85	73.8	1005	1004.7	19	5.2	13

Tabel 13. Lanjutan.

TANGGAL	TEMP (°F)	DEWP (°F)	SLP (Milibar)	STP (Milibar)	VISIB (Mil)	MXSPD (Knot)	WDSP (Knot)
23-12-16	84	74.8	1005.5	1005.2	17.1	5	8.4
24-12-16	85.9	73.5	1006.2	1005.9	11.1	5.2	6
25-12-16	85.3	74.3	1007.2	1006.8	9.9	5	4.6
26-12-16	83.4	75.4	1007.6	1007.2	15	4.1	6.8
27-12-16	82	76.1	1006.8	1006.4	19	4.1	6.9
28-12-16	83	75.9	1008	1007.7	8.9	4.6	3.6
29-12-16	82.9	75.9	1008	1007.7	9.9	4.8	5.1
30-12-16	78.9	75.5	1008.7	1008.3	13	3.2	5.2
31-12-16	82.3	75.7	1008.2	1007.9	8	4.3	3.9

Keterangan :

TEMP : Suhu udara rata-rata harian (°F)

DEWP : Titik embun rata-rata harian (°F)

SLP : Tekanan udara rata-rata harian di atas permukaan laut (Milibar)

STP : Tekanan udara rata-rata harian di stasiun cuaca (Milibar)

MXSPD : Kecepatan angin tertinggi dalam satu hari (Knot)

VISIB : Jarak pandang atau visibilitas rata-rata harian (Mil)

WDSP : Kecepatan angin rata-rata harian (Knot)

Sumber :

National Climatic Data Center, U.S. Department of Commerce.
(www7.ncdc.noaa.gov/CDO/dataproduct)

Halaman ini sengaja dikosongkan

Validasi Hasil Prediktor Cuaca Menggunakan Metode JST-PSO untuk prediksi cuaca rata-rata satu hari berikutnya dan satu jam berikutnya dapat dilihat pada tabel dibawah ini.

- Prediksi Cuaca Rata-Rata Satu Hari Berikutnya

Tabel 14. Prediksi Satu Hari Berikutnya

Tanggal	Kecepatan Angin (Knot)		Visibilitas (Mil)	
	Prediksi	Aktual	Prediksi	Aktual
2-12-2016	4.67268	4.9	4.8549	3.5
10-12-2016	4.64691	4.9	4.83752	5.1
21-12-2016	6.7942	11.1	5.00552	5.1
29-12-2016	5.0412	5.1	4.886	4.8

- Prediksi Cuaca Rata-Rata Satu Jam Berikutnya

Tabel 15. Prediksi Satu Jam Berikutnya Tanggal 20 Juni 2017

Jam	Kecepatan Angin (Knot)		Visibilitas (Mil)	
	Prediksi	Aktual	Prediksi	Aktual
12.00 WIB	5.2737	9	5.21953	6.2
14.00 WIB	5.54356	8	5.30339	6.2
16.00 WIB	6.56394	7	5.6829	6.2

Lampiran 11. Spesifikasi Arsitektur JST

Spesifikasi arsitektur dari metode gabungan Jaringan Saraf Tiruan dan *Particle Swarm Optimization* untuk melakukan prediksi cuaca, yaitu sebagai berikut:

a. Kecepatan Angin

- Arsitektur JST : Pada Gambar 4.15 halaman 61
 - Node masukan berjumlah 4 yang meliputi titik embun, tekanan udara di permukaan laut, tekanan udara di stasiun cuaca, dan kecepatan angin maksimum.
 - Node tersembunyi yang terbaik berjumlah 6.
 - Node keluaran berjumlah 1 yaitu kecepatan angin.
- Fungsi Aktivasi JST: Tangen Sigmoid
- Jumlah Partikel PSO : 40
- Iterasi PSO : 15
- Koefisien c_1 : 1,5
- Koefisien c_2 : 2
- Jumlah Bobot : 37
- Bobot Optimum JST : Pada Tabel 4.13 halaman 69

b. Visibilitas

- Arsitektur JST : Pada gambar 4.19 halaman 65
 - Node masukan berjumlah 4 yang meliputi suhu udara, titik embun, tekanan udara di permukaan laut, dan tekanan udara di stasiun cuaca.
 - Node tersembunyi yang terbaik berjumlah 9.
 - Node keluaran berjumlah 1 yaitu kecepatan angin.
- Fungsi Aktivasi JST: Tangen Sigmoid
- Jumlah Partikel PSO : 40
- Iterasi PSO : 15
- Koefisien c_1 : 1,5
- Koefisien c_2 : 2
- Jumlah Bobot : 55
- Bobot Optimum JST : Pada Tabel 4.14 halaman 70

Spesifikasi arsitektur dari metode Jaringan Saraf Tiruan untuk melakukan prediksi cuaca, yaitu sebagai berikut:

a. Kecepatan Angin

- Arsitektur JST : Pada Gambar 4.7 halaman 51
 - Node masukan berjumlah 4 yang meliputi titik embun, tekanan udara di permukaan laut, tekanan udara di stasiun cuaca, dan kecepatan angin maksimum.
 - Node tersembunyi yang terbaik berjumlah 3.
 - Node keluaran berjumlah 1 yaitu kecepatan angin.
- Fungsi Aktivasi JST: Tangen Sigmoid
- Epoch : 150
- *Learning Rate* : 0.1
- Koefisien Kombinasi : 1
- Jumlah Bobot : 19
- Bobot Optimum JST : Pada Tabel 4.11 halaman 67

b. Visibilitas

- Arsitektur JST : Pada gambar 4.11 halaman 55
 - Node masukan berjumlah 4 yang meliputi suhu udara, titik embun, tekanan udara di permukaan laut, dan tekanan udara di stasiun cuaca.
 - Node tersembunyi yang terbaik berjumlah 4.
 - Node keluaran berjumlah 1 yaitu kecepatan angin.
- Fungsi Aktivasi JST: Tangen Sigmoid
- Epoch : 150
- *Learning Rate* : 0.1
- Koefisien Kombinasi : 1
- Jumlah Bobot : 25
- Bobot Optimum JST : Pada Tabel 4.12 halaman 68